Bundesamt
für Sicherheit in der
Informationstechnik

SYS.1.6 Containerisierung

1. Beschreibung

1.1. Einleitung

Der Begriff *Containerisierung* bezeichnet ein Konzept, bei dem Ressourcen eines Betriebssystems partitioniert werden, um Ausführungsumgebungen für Prozesse zu schaffen. Hierbei können je nach verwendetem Betriebssystem unterschiedliche Techniken zum Einsatz kommen, die sich in Funktionsumfang und Sicherheitseigenschaften unterscheiden. Oft wird auch von einer „Betriebssystemvirtualisierung" gesprochen. Es wird jedoch kein vollständiges Betriebssystem virtualisiert, sondern es werden lediglich bestimmte Ressourcen durch einen geteilten Kernel zur Verfügung gestellt. Allgemein wird der Begriff *Container* verwendet, um das entstehende Konstrukt zu bezeichnen.

Bevor leichtfertig komplexe Container-Umgebungen gebaut und verwendet werden, sollte zunächst gründlich abgewogen werden, ob der Aufwand für die Erstellung und den Betrieb einer Container-Umgebungen in einem geeigneten Verhältnis zum tatsächlichen Nutzen steht. Der sachgerechte Betrieb von Container-Umgebungen ist sehr komplex und es sind viele Anforderungen zu beachten. Container können genutzt werden, um als zusätzlicher Trennungsmechanismus eine Härtung der Umgebung zu erreichen, sofern die Art der Containertechnik und die vorgenommene Konfiguration hierfür sachgerecht und geeignet sind.

Es findet eine Unterscheidung zwischen Applikations-Containern (z. B. nach der Spezifikation der Open Container Initiative / OCI) und System-Containern statt. System-Container sind der älteste Typ von Containern, wie z. B. FreeBSD Jails, Solaris Zones, OpenVZ, LXC und LXD. Sie stellen eine Umgebung zur Verfügung, die sich ähnlich einem eigenständigen Betriebssystem verhält, entsprechende Dienste anbieten und mehrere Anwendungen ausführen kann. Applikations-Container hingegen sind speziell für den Fall gedacht, eine einzelne Anwendung auszuführen. Sie folgen dem Lebenszyklus der Anwendung, bieten aber auch innerhalb des Containers keine betriebssystemspezifischen Dienste an. Aus technischer Sicht setzen diese beiden Typen jedoch auf den gleichen Mechanismus zur Trennung, also die Prozessisolation durch den Kernel.

Beim Beispiel Linux-Container (LXC) werden dazu hauptsächlich die Mechanismen *Namespaces* und *cgroups* verwendet, um die Prozessisolation noch zu ergänzen.

- Über Namespaces wird kontrolliert, welche Ressourcen ein Prozess sehen kann. Es gibt sieben unterschiedliche Namespaces: Mount (mnt), Prozess-ID (pid), Netz (net), Interprozess-Kommunikation (ipc), UTS (uts), User ID (user) und Control Group (cgroup).
- Über cgroups wird kontrolliert, welche Ressourcen bzw. welchen Anteil einer Ressource ein Prozess benutzen kann. Zu Ressourcen zählen insbesondere Memory, CPU, BLKIO oder PIDS.

Bei der Verwendung von Namespaces und cgroups sind viele Einzelheiten zu beachten. Unter anderem spielt die Reihenfolge, in der Namespaces geteilt werden, eine entscheidende Rolle. Für diesen Zweck wurden Container-Runtimes entwickelt, wie etwa *runc*, *crun*, *railcar* oder auch *katacontainers*. Die Hauptaufgabe von Container-Runtimes ist das Erstellen einer besonderen Ausführungsumgebung für Prozesse. Sie kommunizieren mit dem Kernel und rufen Syscalls mit den entsprechenden Parametern sowie in der korrekten Reihenfolge auf, um die gewünschte Ausführungsumgebung zu erhalten.

Üblicherweise benötigen Container analog zu einem Betriebssystem ein Dateisystem, in dem die auszuführenden Programme abgelegt werden. Im Container-Umfeld haben sich bestimmte Dateiformate etabliert, um diese Dateisysteme zu beschreiben. Diese werden als *Image,* fälschlicherweise aber teilweise ebenfalls als Container bezeichnet. Abhängig vom verwendeten Container-Typ kann der Inhalt dieser Images von einer einzelnen statisch kompilierten Anwendung bis hin zum vollständigen Inhalt eines Betriebssystems inklusive diverser Ausführungsumgebun-

gen und sonstiger Abhängigkeiten reichen. Images sind transportable, abgeschlossene Einheiten, die in einer Container-Umgebung ausgebracht werden können und alle Komponenten für die Funktionsfähigkeit enthalten.

Neben der Runtime gibt es Container-Engines wie z. B. *Docker*, *Rocket* oder *CRI-O*, die viele Verwaltungsaufgaben übernehmen. In erster Linie bilden sie die Schnittstelle zu Benutzenden und verarbeiten übergebene Befehle. Sie sorgen dafür, dass die benötigten Images zur Verfügung stehen und entsprechende Metadaten vorbereitet sind. Schließlich ruft die Container-Engine die Container-Runtime mit entsprechenden Parametern auf. Die Container-Engine ist damit nicht Teil des Mechanismus Containerisierung, sondern übernimmt hier eine Verwaltungsfunktion.

Weiterhin gibt es unterschiedliche Arten von Containern. Diese unterscheiden sich anhand des Einsatzszenarios und des Lebenszyklus eines Containers. Als *persistenter* Container wird ein Container bezeichnet, der für einen längeren Einsatz gedacht ist. Es kann durchaus valide Gründe geben, in Containern dauerhaft Daten zu speichern. Besonders im Cloud-Umfeld sind jedoch häufig *volatile* Container anzutreffen. Dort haben Container in der Regel eine viel kürzere Lebensdauer, die zudem oft von Orchestrierungswerkzeugen bestimmt wird.

Aus der OCI gibt es Bestrebungen, Standards und Referenzimplementierung zur Verfügung zu stellen. Beispielsweise ist *runc* die Standard-Referenzimplementierung einer Container-Runtime. Andere Container-Runtimes, die kompatibel zum OCI-Standard sind, können somit weitgehend ausgetauscht und verwendet werden.

1.2. Zielsetzung

Ziel dieses Bausteins ist der Schutz von Informationen, die in, von oder mit Containern verarbeitet, angeboten oder darüber übertragen werden. Der Baustein behandelt, wie Container grundsätzlich abgesichert werden können. Dabei wird unterschieden zwischen den Diensten zum Betrieb der Container, also der Software, die für Konfiguration und Verwaltung der Container zuständig ist, sowie den Applikationen und Diensten, die innerhalb der Container ausgeführt werden.

1.3. Abgrenzung und Modellierung

Der Baustein SYS.1.6 *Containerisierung* ist immer anzuwenden, sobald Dienste oder Anwendungen in Containern betrieben werden.

Dieser Baustein betrachtet Container unabhängig von konkreten Produkten. Die Anforderungen orientieren sich an den Fähigkeiten derzeit am Markt verfügbarer Implementierungen. Bei der Produktauswahl ist der Baustein APP.6 *Allgemeine Software* zu berücksichtigen.

Der Baustein ergänzt die Aspekte, die in den Bausteinen SYS.1.1 *Allgemeiner Server* und SYS.1.3 *Server unter Linux und Unix* behandelt werden, um Spezifika von Containerisierung. Die Anforderungen dieser Bausteine sollten vom Host-System erfüllt werden, unabhängig davon, ob dieses auf physischen Servern ausgeführt wird oder virtualisiert ist. Weiterhin gelten die Anforderungen dieser Bausteine ebenfalls für jede der im Rahmen der Containerisierung erzeugten Userspace-Umgebung. Dabei müssen regelmäßig weitere Bausteine wie z. B. CON.8 *Software-Entwicklung* oder die Bausteine der Teilschicht OPS.1.1 *Kern-IT-Betrieb* berücksichtigt werden. Dies trifft vor allem auf die Erstellung von Images zu.

Typischerweise kommunizieren Container durch virtuelle Netze auf dem Host-System miteinander. Die Bausteine der Teilschichten NET.1 *Netze* und NET.3 *Netzkomponenten* müssen entsprechend berücksichtigt werden.

Sicherheitsanforderungen möglicher Dienste, wie z. B. Webserver (APP.3.2 *Webserver*) oder Webanwendungen (APP.3.1 *Webanwendungen und Webservices*) sind Gegenstand eigener Bausteine, die zusätzlich anzuwenden sind. Sollte das Host-System virtualisiert sein, ist der Baustein SYS.1.5 *Virtualisierung* zu modellieren.

Sollten die Container und die darunterliegende Infrastruktur nicht vollständig selber betrieben und alleinig genutzt werden, sondern werden Teile hiervon durch Dritte bereitgestellt oder von Dritten genutzt, sind zusätzliche Anforderungen der Bausteine OPS.2.3 *Nutzung von Outsourcing* und OPS.2.2 *Cloud-Nutzung* sowie OPS.3.2 *Anbieten von Outsourcing* zu berücksichtigen.

Der Baustein enthält grundsätzliche Anforderungen zur Einrichtung und zum Betrieb von Containerisierung. Die weiteren im Themenumfeld üblichen Dienste, wie z. B. Orchestrierung von Containern, Speichersysteme, virtuelle Netze, Automatisierung für CI/CD-Pipelines oder der Betrieb von Image-Registries, werden hier nicht betrachtet. Ebenso wenig trifft dieser Bausteine Aussagen zu Anforderungen, die für den Bau von Images gelten. Für Anforderungen zur Orchestrierung von Containern mit Kubernetes ist der Baustein APP.4.4 *Kubernetes* zu betrachten.

2. Gefährdungslage

Da IT-Grundschutz-Bausteine nicht auf individuelle Informationsverbünde eingehen können, werden zur Darstellung der Gefährdungslage typische Szenarien zugrunde gelegt. Die folgenden spezifischen Bedrohungen und Schwachstellen sind für den Baustein SYS.1.6 *Containerisierung* von besonderer Bedeutung.

2.1. Schwachstellen oder Schadsoftware in Images

Container werden vorrangig auf Basis von vorgefertigten Images erstellt, die selbst erstellt, aber auch häufig aus dem Internet bezogen werden. Des Weiteren wird Software zunehmend in Form von Images ausgeliefert. Diese Images kann der IT-Betrieb auch für die Erstellung seiner eigenen Images nutzen, indem er Software oder Konfigurationen ergänzt, verändert oder auch entfernt.

Die in den Images enthaltene Software könnte verwundbar und die aus dem Image gestarteten Container könnten dadurch angreifbar sein. Solche Schwachstellen können dem IT-Betrieb auch häufig nicht bekannt sein, da die in den Images enthaltene Software oft nicht in der eigenen Software-Verwaltung erfasst ist. Der IT-Betrieb muss sich grundsätzlich darauf verlassen, dass Updates über den Prozess der Image-Erstellung verfügbar gemacht werden. Von außen betrachtet ist häufig nur aufwendig zu ermitteln, welche Software-Pakete in diesen Images vorhanden sind.

Zudem könnten Images absichtlich integrierte Schadsoftware enthalten, wie z. B. Ransomware oder Kryptominer. Da ein einzelnes Image oft in einer großen Anzahl von Containern ausgebracht wird, kann der resultierende Schaden immens sein.

2.2. Administrative Zugänge ohne Absicherung

Um Container-Dienste auf einem Host zu verwalten, werden administrative Zugänge benötigt. Diese Zugänge sind oft als Dienste realisiert, die entweder lokal oder über das Netz angesprochen werden können. Mechanismen zur Authentisierung und Verschlüsselung der administrativen Zugänge sind häufig vorhanden, aber nicht bei allen Produkten standardmäßig aktiviert.

Wenn Unbefugte auf die Netzsockets oder das Host-System zugreifen können, können sie über ungeschützte administrative Zugänge Befehle ausführen, die zum Verlust der Vertraulichkeit, Integrität und Verfügbarkeit aller auf diesem Host ausgeführten Container führen können.

2.3. Ressourcenkonflikte auf dem Host

Einzelne Container können den Host überlasten und so die Verfügbarkeit aller anderen Container auf dem Host oder auch den Betrieb des Host-Systems selbst gefährden.

2.4. Unberechtigte Kommunikation

Alle Container auf einem Host sind grundsätzlich in der Lage, miteinander, mit dem Host sowie beliebigen anderen Hosts zu kommunizieren. Sofern diese Kommunikation nicht eingeschränkt wird, kann dies ausgenutzt werden, um z. B. andere Container oder Hosts anzugreifen.

Weiterhin besteht die Gefahr, dass Container von außen erreichbar sind, auch wenn dies nicht erwünscht ist. So könnte ein Angriff gegen Dienste, die eigentlich nur intern erreichbar sein sollten, von außen erfolgen. Diese Gefährdung erhöht sich durch die geringere Aufmerksamkeit, die solchen internen Diensten oft entgegengebracht wird. Wird z. B. eine Verwundbarkeit auf einem nur intern eingesetzten Dienst toleriert und ist dieser auch von außen erreichbar, kann dies den gesamten Betrieb erheblich gefährden.

2.5. Ausbruch aus dem Container auf das Host-System

Besteht die Möglichkeit, im Container eigenen Code auszuführen, kann bei einem Angriff möglicherweise die Isolation des Containers gegenüber anderen Containern oder dem Host überwunden und auf andere Container, das Host-System oder die Infrastruktur zugegriffen werden. Es wird auch von einem „Container-Ausbruch“ gesprochen. Dieser Angriff kann z. B. über Schwachstellen in Prozessoren, im Betriebssystem-Kernel oder in lokal angebotenen Infrastruktur-Diensten wie z. B. DNS oder SSH erfolgen.

Somit könnte bei einem Angriff die Kontrolle über das Host-System oder andere Systeme aus der Infrastruktur übernommen werden. Es droht der Verlust der Vertraulichkeit, Integrität und Verfügbarkeit aller auf diesem Host ausgeführten Container sowie auf dem Host selbst, falls dort ebenfalls erhöhte Privilegien erlangt werden können.

2.6. Datenverluste durch das Container-Management

Im Rahmen der Verwaltung von Containern können diese ausgeschaltet werden, ohne darin gerade ausgeführter Software die Gelegenheit zu geben, z. B. aktuelle Schreibprozesse abzuschließen (nicht ordnungsgemäßes Herunterfahren). Sollten zu diesen Zeitpunkten Daten durch den Container verarbeitet werden, sind alle diese Daten verloren. Auch Daten, die persistent im Container gespeichert sind, können so dauerhaft verloren gehen.

2.7. Vertraulichkeitsverlust von Zugangsdaten

Die Prinzipien des Aufbaus und der Erstellung von Images für Container setzen voraus, dass Zugangsdaten, z. B. für Datenbanken, im Container verfügbar sind. Über die Images selbst, die Skripte zur Erstellung der Images oder die Versionskontrolle der Skripte können solche Zugangsdaten in unbefugte Hände gelangen.

Oft werden Zugangsdaten auch zum Erstellungszeitpunkt des Containers als Umgebungsvariable verfügbar gemacht. Hier droht ebenfalls der Vertraulichkeitsverlust dieser Daten.

2.8. Ungeordnete Bereitstellung und Verteilung von Images

Im Gegensatz zu klassischen Installationen durch den IT-Betrieb, wo die Kontrolle über die ausgebrachten Anwendungen, Komponenten und Diensten vollständig beim IT-Betrieb selbst liegt, geht diese bei z. B. bei Automatisierung durch CI/CD in Container-Umgebungen verloren. Vielmehr stellt der IT-Betrieb nur eine Plattform bereit, in die Entwickelnde direkt ihre Anwendungen inklusive sämtlicher Abhängigkeiten einbringen und jederzeit verändern können.

3. Anforderungen

Im Folgenden sind die spezifischen Anforderungen des Bausteins SYS.1.6 *Containerisierung* aufgeführt. Der oder die Informationssicherheitsbeauftragte (ISB) ist dafür zuständig, dass alle Anforderungen gemäß dem festgelegten Sicherheitskonzept erfüllt und überprüft werden. Bei strategischen Entscheidungen ist der oder die ISB stets einzubeziehen.

Im IT-Grundschutz-Kompendium sind darüber hinaus weitere Rollen definiert. Sie sollten besetzt werden, insofern dies sinnvoll und angemessen ist.

Zuständigkeiten	Rollen
Grundsätzlich zuständig	IT-Betrieb
Weitere Zuständigkeiten	Keine

Genau eine Rolle sollte *Grundsätzlich zuständig* sein. Darüber hinaus kann es noch *Weitere Zuständigkeiten* geben. Falls eine dieser weiteren Rollen für die Erfüllung einer Anforderung vorrangig zuständig ist, dann wird diese Rolle hinter der Überschrift der Anforderung in eckigen Klammern aufgeführt. Die Verwendung des Singulars oder Plurals sagt nichts darüber aus, wie viele Personen diese Rollen ausfüllen sollen.

3.1. Basis-Anforderungen

Die folgenden Anforderungen MÜSSEN für diesen Baustein vorrangig erfüllt werden.

SYS.1.6.A1 Planung des Container-Einsatzes (B)

Bevor Container eingesetzt werden, MUSS zunächst das Ziel des Container-Einsatzes (z. B. Skalierung, Verfügbarkeit, Wegwerf-Container zur Sicherheit oder CI/CD) festgelegt werden, damit alle sicherheitsrelevanten Aspekte der Installation, des Betriebs und der Außerbetriebnahme geplant werden können. Bei der Planung SOLLTE auch der Betriebsaufwand berücksichtigt werden, der durch Container-Einsatz oder Mischbetrieb entsteht. Die Planung MUSS angemessen dokumentiert werden.

SYS.1.6.A2 Planung der Verwaltung von Containern (B)

Die Verwaltung der Container DARF NUR nach einer geeigneten Planung erfolgen. Diese Planung MUSS den gesamten Lebenszyklus von Inbetrieb- bis Außerbetriebnahme inklusive Betrieb und Updates umfassen. Bei der Planung der Verwaltung MUSS berücksichtigt werden, dass Erstellende eines Containers aufgrund der Auswirkungen auf den Betrieb in Teilen wie Administrierende zu betrachten sind.

Start, Stopp und Überwachung der Container MUSS über die eingesetzte Verwaltungssoftware erfolgen.

SYS.1.6.A3 Sicherer Einsatz containerisierter IT-Systeme (B)

Bei containerisierten IT-Systemen MUSS berücksichtigt werden, wie sich eine Containerisierung auf die betriebenen IT-Systeme und Anwendungen auswirkt, dies betrifft insbesondere die Verwaltung und Eignung der Anwendungen.

Es MUSS anhand des Schutzbedarfs der Anwendungen geprüft werden, ob die Anforderungen an die Isolation und Kapselung der containerisierten IT-Systeme und der virtuellen Netze sowie der betriebenen Anwendungen hinreichend erfüllt sind. In diese Prüfung SOLLTEN die betriebssystemeigenen Mechanismen mit einbezogen werden. Für die virtuellen Netze nimmt der Host die Funktion einer Netzkomponente wahr, die Bausteine der Teilschichten NET.1 *Netze* und NET.3 *Netzkomponenten* MÜSSEN entsprechend berücksichtigt werden. Logische und Overlay-Netze MÜSSEN ebenfalls betrachtet und modelliert werden. Weiterhin MÜSSEN die eingesetzten containerisierten IT-Systeme den Anforderungen an die Verfügbarkeit und den Datendurchsatz genügen.

Im laufenden Betrieb SOLLTEN die Performance und der Zustand der containerisierten IT-Systeme überwacht werden (sogenannte Health Checks).

SYS.1.6.A4 Planung der Bereitstellung und Verteilung von Images (B)

Der Prozess zur Bereitstellung und Verteilung von Images MUSS geplant und angemessen dokumentiert werden.

SYS.1.6.A5 Separierung der Administrations- und Zugangsnetze bei Containern (B)

Die Netze für die Administration des Hosts, die Administration der Container und deren Zugangsnetze MÜSSEN dem Schutzbedarf angemessen separiert werden. Grundsätzlich SOLLTE mindestens die Administration des Hosts nur aus dem Administrationsnetz möglich sein.

Es SOLLTEN nur die für den Betrieb notwendigen Kommunikationsbeziehungen erlaubt werden.

SYS.1.6.A6 Verwendung sicherer Images (B)

Es MUSS sichergestellt sein, dass sämtliche verwendeten Images nur aus vertrauenswürdigen Quellen stammen. Es MUSS eindeutig identifizierbar sein, wer das Image erstellt hat.

Die Quelle MUSS danach ausgewählt werden, dass die im Image enthaltene Software regelmäßig auf Sicherheitsprobleme geprüft wird und diese behoben sowie dokumentiert werden. Die Quelle MUSS diesen Umgang mit Sicherheitsproblemen zusichern und einhalten.

Die verwendete Version von Basis-Images DARF NICHT abgekündigt („deprecated“) sein. Es MÜSSEN eindeutige Versionsnummern angegeben sein. Wenn ein Image mit einer neueren Versionsnummer verfügbar ist, MUSS im Rahmen des Patch- und Änderungsmanagement geprüft werden, ob und wie dieses ausgerollt werden kann.

SYS.1.6.A7 Persistenz von Protokollierungsdaten der Container (B)

Die Speicherung der Protokollierungsdaten der Container MUSS außerhalb des Containers, mindestens auf dem Container-Host, erfolgen.

SYS.1.6.A8 Sichere Speicherung von Zugangsdaten bei Containern (B)

Zugangsdaten MÜSSEN so gespeichert und verwaltet werden, dass nur berechtigte Personen und Container darauf zugreifen können. Insbesondere MUSS sichergestellt sein, dass Zugangsdaten nur an besonders geschützten Orten und nicht in den Images liegen. Die von der Verwaltungssoftware des Container-Dienstes bereitgestellten Verwaltungsmechanismen für Zugangsdaten SOLLTEN eingesetzt werden.

Mindestens die folgenden Zugangsdaten MÜSSEN sicher gespeichert werden:

- Passwörter jeglicher Accounts,
- API-Keys für von der Anwendung genutzte Dienste,
- Schlüssel für symmetrische Verschlüsselungen sowie
- private Schlüssel bei Public-Key-Authentisierung.

3.2. Standard-Anforderungen

Gemeinsam mit den Basis-Anforderungen entsprechen die folgenden Anforderungen dem Stand der Technik für diesen Baustein. Sie SOLLTEN grundsätzlich erfüllt werden.

SYS.1.6.A9 Eignung für Container-Betrieb (S)

Die Anwendung oder der Dienst, die oder der im Container betrieben werden soll, SOLLTE für den Container-Betrieb geeignet sein. Dabei SOLLTE berücksichtigt werden, dass Container häufiger für die darin ausgeführte Anwendung unvorhergesehen beendet werden können. Die Ergebnisse der Prüfung nach SYS.1.6.A3 *Sicherer Einsatz containerisierter IT-Systeme* SOLLTE nachvollziehbar dokumentiert werden.

SYS.1.6.A10 Richtlinie für Images und Container-Betrieb (S)

Es SOLLTE eine Richtlinie erstellt und angewendet werden, die die Anforderungen an den Betrieb der Container und die erlaubten Images festlegt. Die Richtlinie SOLLTE auch Anforderungen an den Betrieb und die Bereitstellung der Images enthalten.

SYS.1.6.A11 Nur ein Dienst pro Container (S)

Jeder Container SOLLTE jeweils nur einen Dienst bereitstellen.

SYS.1.6.A12 Verteilung sicherer Images (S)

Es SOLLTE angemessen dokumentiert werden, welche Quellen für Images als vertrauenswürdig klassifiziert wurden und warum. Zusätzlich SOLLTE der Prozess angemessen dokumentiert werden, wie Images bzw. die im Image enthaltenen Softwarebestandteile aus vertrauenswürdigen Quellen bezogen und schließlich für den produktiven Betrieb bereitgestellt werden.

Die verwendeten Images SOLLTEN über Metadaten verfügen, die die Funktion und die Historie des Images nachvollziehbar machen. Digitale Signaturen SOLLTEN jedes Image gegen Veränderung absichern.

SYS.1.6.A13 Freigabe von Images (S)

Alle Images für den produktiven Betrieb SOLLTEN wie Softwareprodukte einen Test- und Freigabeprozess gemäß dem Baustein OPS.1.1.6 *Software-Test und Freigaben* durchlaufen.

SYS.1.6.A14 Aktualisierung von Images (S)

Bei der Erstellung des Konzeptes für das Patch- und Änderungsmanagement gemäß OPS.1.1.3 *Patch- und Änderungsmanagement* SOLLTE entschieden werden, wann und wie die Updates der Images bzw. der betriebenen Software oder des betriebenen Dienstes ausgerollt werden. Bei persistenten Containern SOLLTE geprüft werden, ob in Ausnahmefällen ein Update des jeweiligen Containers geeigneter ist, als den Container vollständig neu zu provisionieren.

SYS.1.6.A15 Limitierung der Ressourcen pro Container (S)

Für jeden Container SOLLTEN Ressourcen auf dem Host-System, wie CPU, flüchtiger und persistenter Speicher sowie Netzbandbreite, angemessen reserviert und limitiert werden. Es SOLLTE definiert und dokumentiert sein, wie das System im Fall einer Überschreitung dieser Limitierungen reagiert.

SYS.1.6.A16 Administrativer Fernzugriff auf Container (S)

Administrative Zugriffe von einem Container auf den Container-Host und umgekehrt SOLLTEN prinzipiell wie administrative Fernzugriffe betrachtet werden. Aus einem Container SOLLTEN KEINE administrativen Fernzugriffe auf den Container-Host erfolgen. Applikations-Container SOLLTEN keine Fernwartungszugänge enthalten. Administrative Zugriffe auf Applikations-Container SOLLTEN immer über die Container-Runtime erfolgen.

SYS.1.6.A17 Ausführung von Containern ohne Privilegien (S)

Die Container-Runtime und alle instanziierten Container SOLLTEN nur von einem nicht-privilegierten System-Account ausgeführt werden, der über keine erweiterten Rechte für den Container-Dienst und das Betriebssystem des Host-Systems verfügt oder diese Rechte erlangen kann. Die Container-Runtime SOLLTE durch zusätzliche Maßnahmen gekapselt werden, etwa durch Verwendung der Virtualisierungserweiterungen von CPUs.

Sofern Container ausnahmsweise Aufgaben des Host-Systems übernehmen sollen, SOLLTEN die Privilegien auf dem Host-System auf das erforderliche Minimum begrenzt werden. Ausnahmen SOLLTEN angemessen dokumentiert werden.

SYS.1.6.A18 Accounts der Anwendungsdienste (S)

Die System-Accounts innerhalb eines Containers SOLLTEN keine Berechtigungen auf dem Host-System haben. Wo aus betrieblichen Gründen diese Berechtigung notwendig ist, SOLLTE diese nur für unbedingt notwendige Daten und Systemzugriffe gelten. Der Account im Container, der für diesen Datenaustausch notwendig ist, SOLLTE im Host-System bekannt sein.

SYS.1.6.A19 Einbinden von Datenspeichern in Container (S)

Die Container SOLLTEN nur auf die für den Betrieb notwendigen Massenspeicher und Verzeichnisse zugreifen können. Nur wenn Berechtigungen benötigt werden, SOLLTEN diese explizit vergeben werden. Sofern die Container-Runtime für einen Container lokalen Speicher einbindet, SOLLTEN die Zugriffsrechte im Dateisystem auf den Service-Account des Containers eingeschränkt sein. Werden Netzspeicher verwendet, so SOLLTEN die Berechtigungen auf dem Netzspeicher selbst gesetzt werden.

SYS.1.6.A20 Absicherung von Konfigurationsdaten (S)

Die Beschreibung der Container-Konfigurationsdaten SOLLTE versioniert erfolgen. Änderungen SOLLTEN nachvollziehbar dokumentiert sein.

3.3. Anforderungen bei erhöhtem Schutzbedarf

Im Folgenden sind für diesen Baustein exemplarische Vorschläge für Anforderungen aufgeführt, die über dasjenige Schutzniveau hinausgehen, das dem Stand der Technik entspricht. Die Vorschläge SOLLTEN bei erhöhtem Schutzbedarf in Betracht gezogen werden. Die konkrete Festlegung erfolgt im Rahmen einer individuellen Risikoanalyse.

SYS.1.6.A21 Erweiterte Sicherheitsrichtlinien (H)

Erweiterte Richtlinien SOLLTEN die Berechtigungen der Container einschränken. Mandatory Access Control (MAC) oder eine vergleichbare Technik SOLLTE diese Richtlinien erzwingen. Die Richtlinien SOLLTEN mindestens folgende Zugriffe einschränken:

- eingehende und ausgehende Netzverbindungen,
- Dateisystem-Zugriffe und
- Kernel-Anfragen (Syscalls).

Die Runtime SOLLTE die Container so starten, dass der Kernel des Host-Systems alle nicht von der Richtlinie erlaubten Aktivitäten der Container verhindert (z. B. durch die Einrichtung lokaler Paketfilter oder durch Entzug von Berechtigungen) oder zumindest Verstöße geeignet meldet.

SYS.1.6.A22 Vorsorge für Untersuchungen (H)

Um Container im Bedarfsfall für eine spätere Untersuchung verfügbar zu haben, SOLLTE ein Abbild des Zustands nach festgelegten Regeln erstellt werden.

SYS.1.6.A23 Unveränderlichkeit der Container (H)

Container SOLLTEN ihr Dateisystem während der Laufzeit nicht verändern können. Dateisysteme SOLLTEN nicht mit Schreibrechten eingebunden sein.

SYS.1.6.A24 Hostbasierte Angriffserkennung (H)

Das Verhalten der Container und der darin betriebenen Anwendungen und Dienste SOLLTE überwacht werden. Abweichungen von einem normalen Verhalten SOLLTEN bemerkt und gemeldet werden. Die Meldungen SOLLTEN im zentralen Prozess zur Behandlung von Sicherheitsvorfällen angemessen behandelt werden.

Das zu überwachende Verhalten SOLLTE mindestens umfassen:

- Netzverbindungen,
- erstellte Prozesse,
- Dateisystem-Zugriffe und
- Kernel-Anfragen (Syscalls).

SYS.1.6.A25 Hochverfügbarkeit von containerisierten Anwendungen (H)

Bei hohen Verfügbarkeitsanforderungen der containerisierten Anwendungen SOLLTE entschieden werden, auf welcher Ebene die Verfügbarkeit realisiert werden soll (z. B. redundant auf der Ebene des Hosts).

SYS.1.6.A26 Weitergehende Isolation und Kapselung von Containern (H)

Wird eine weitergehende Isolation und Kapselung von Containern benötigt, dann SOLLTEN folgende Maßnahmen nach steigender Wirksamkeit geprüft werden:

- feste Zuordnung von Containern zu Container-Hosts,
- Ausführung der einzelnen Container und/oder des Container-Hosts mit Hypervisoren,
- feste Zuordnung eines einzelnen Containers zu einem einzelnen Container-Host.

4. Weiterführende Informationen

4.1. Wissenswertes

Weiterführende Informationen zu Gefährdungen und Sicherheitsmaßnahmen im Bereich Container finden sich unter anderem in folgenden Veröffentlichungen:

- NIST 800-190
  https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-190.pdf
- CIS Benchmark Docker
  https://www.cisecurity.org/benchmark/docker/
- OCI – Open Container Initiative
  https://www.opencontainers.org/
- CNCF – Cloud Native Computing Foundation
  https://www.cncf.io/
- SANS Checkliste
  https://www.sans.org/reading-room/whitepapers/auditing/checklist-audit-docker-containers-37437
- Docker Security Guide
  https://docs.docker.com/engine/security/

Bundesamt
für Sicherheit in der
Informationstechnik

SYS.1.7 IBM Z

1. Beschreibung

1.1. Einleitung

Systeme vom Typ „IBM Z" gehören zu den Server-Systemen, die allgemein als Großrechner („Mainframes") bezeichnet werden. Großrechner haben sich von klassischen Einzelsystemen mit Stapelverarbeitung hin zu modernen Client-Server-Systemen entwickelt. Die Z-Architektur ist der Nachfolger der 1964 eingeführten S/360-Architektur und wird bei heutigen Großrechner-Installationen häufig eingesetzt.

1.2. Zielsetzung

Ziel dieses Bausteins ist der Schutz von Informationen, die auf Z-Systemen verarbeitet, angeboten oder darüber übertragen werden.

1.3. Abgrenzung und Modellierung

Der Baustein SYS.1.7 *IBM Z* ist auf jeden Server anzuwenden, der auf der Z-Architektur von IBM basiert.

Der Baustein SYS.1.1 *Allgemeiner Server* bildet die Grundlage für die Absicherung von Servern. Für Z-Systeme müssen sowohl die dort aufgeführten allgemeinen Anforderungen als auch die konkreten Anforderungen im vorliegenden Baustein erfüllt werden.

Für die Z-Hardware sind verschiedene Betriebssysteme verfügbar (z. B. z/OS, z/VM, KVM oder Linux on Z). Die Auswahl erfolgt in der Regel anhand der Parameter Rechnergröße und Einsatzzweck. Die Empfehlungen dieses Bausteins beschränken sich im Wesentlichen auf das Betriebssystem z/OS. Ausgewählte Sicherheitsaspekte von z/VM werden ebenfalls angesprochen. Für das Betriebssystem Linux on Z wird auf den Baustein SYS.1.3 *Server unter Linux und Unix* verwiesen.

Weitere für IBM Z besonders relevante Anforderungen finden sich im Baustein ORP.4 *Identitäts- und Berechtigungsmanagement* sowie in OPS.1.2.5 *Fernwartung*.

Ein wichtiger Bestandteil der Sicherheitskonzeption von Z-Systemen auf technischer Ebene ist das eingesetzte Sicherheitssystem, beispielsweise TopSecret, ACF2 (Access Control Facility) oder RACF (Resource Access Control Facility). Um die Darstellung zu vereinfachen, wird im Folgenden nur auf RACF eingegangen. Die Empfehlungen sind entsprechend anzupassen, wenn ein anderes Sicherheitssystem eingesetzt wird.

Die jeweils spezifischen Dienste, die vom Z-System angeboten werden, sind nicht Bestandteil dieses Bausteins. Für diese Dienste müssen zusätzlich noch weitere Bausteine umgesetzt werden, gemäß den Ergebnissen der IT-Grundschutz-Modellierung.

2. Gefährdungslage

Da IT-Grundschutz-Bausteine nicht auf individuelle Informationsverbünde eingehen können, werden zur Darstellung der Gefährdungslage typische Szenarien zugrunde gelegt. Die folgenden spezifischen Bedrohungen und Schwachstellen sind für den Baustein SYS.1.7 *IBM Z* von besonderer Bedeutung.

2.1. Unzureichende oder fehlerhafte Konfiguration der Hardware oder des z/OS-Betriebssystems

Die Konfiguration eines z/OS-Betriebssystems ist sehr komplex und erfordert an vielen Stellen den Eingriff durch Systemadministrierende. Durch falsche oder unzureichende Konfiguration können Schwachstellen entstehen, die

zu entsprechenden Sicherheitsproblemen führen können. SuperVisor Calls (SVCs) sind beispielsweise Aufrufe zu speziellen z/OS-Dienstprogrammen, die im hoch autorisierten Kernel-Modus laufen. Unsichere SVC-Programme können unter Umständen benutzt werden, um z/OS-Sicherheitsmechanismen zu umgehen.

2.2. Fehlerhafte Konfiguration des z/OS-Sicherheitssystems RACF

In einem z/OS-Betriebssystem ist für die Authentisierung von Benutzenden und deren Autorisierung auf Ressourcen ein spezielles Sicherheitssystem zuständig. Hierfür wird häufig RACF eingesetzt. Die Konfiguration von RACF im Auslieferungszustand entspricht in der Regel nicht den Sicherheitsanforderungen im jeweiligen Einsatzszenario. Die Ressourcen und z/OS-System-Kommandos werden beispielsweise über spezielle Klassen im RACF geschützt. Durch unzureichende Definitionen dieser Klassen ist es möglich, dass Benutzende Systembefehle absetzen können, die unter Umständen den stabilen Systembetrieb beeinträchtigen.

2.3. Fehlbedienung der z/OS-Systemfunktionen

Aufgrund der Komplexität eines z/OS-Betriebssystems und seiner Komponenten können Fehlbedienungen nicht vollständig ausgeschlossen werden. Je nach Art der Fehlbedienung können in der Folge einzelne Komponenten oder das gesamte System ausfallen. Verriegeln sich zum Beispiel Ressourcen gegenseitig (Contention), können bestimmte Funktionen so lange nicht verfügbar sein, bis die Verriegelung wieder gelöst wird. Oft sind eine Reihe von Systemabfragen (Displays) und viel Betriebserfahrung notwendig, um gegenseitige Verriegelungen mithilfe der richtigen z/OS-Kommandos wieder aufzulösen.

2.4. Manipulation der z/OS-Systemsteuerung

z/OS-Systeme lassen sich über vielfältige Schnittstellen beeinflussen, zum Beispiel über die Hardware Management Console, lokale sowie entfernte MCS-Konsolen, Automationsverfahren und Fernwartungszugänge. Wenn beispielsweise der physische oder der logische Zugang zu entfernten MCS-Konsolen unzureichend geschützt ist, können die z/OS-Systeme unter Umständen von dort aus unbefugt manipuliert werden.

2.5. Angriffe über TCP/IP auf z/OS-Systeme

Um ein z/OS-System über die Netzanbindung anzugreifen, sind häufig keine Spezialkenntnisse der Netzarchitektur oder von z/OS erforderlich. Durch die TCP/IP-Anbindung an (unter Umständen öffentliche) Netze und die Unix System Services sind viele z/OS-Systeme über Standardprotokolle und Dienste, wie z. B. HTTP oder FTP, für interne bzw. externe Angriffe erreichbar. Bei externen Angriffen können unter Umständen über die TCP/IP-Anbindung an öffentliche Netze Denial-of-Service-Angriffe gegen die angebotenen Dienste durchgeführt oder übertragene Daten unbefugt gelesen oder manipuliert werden. Interne Angreifende können über die TCP/IP-Anbindung an interne Netze versuchen, ihre Berechtigungen zu erhöhen, indem sie etwa Kennung und Passwort von Benutzenden mit SPECIAL-Rechten ausspähen.

2.6. Fehlerhafte Zeichensatzkonvertierung beim Einsatz von z/OS

EBCDIC, ASCII und Unicode sind Zeichensätze, die festlegen, wie Buchstaben, Ziffern und andere Zeichen mithilfe von Bits dargestellt werden. z/OS-Systeme arbeiten mit EBCDIC-Code. Lediglich HFS- und zFS-Dateisysteme, die unter Unix System Services (USS) eingesetzt werden, lassen sowohl ASCII- als auch EBCDIC-Speicherung zu. Beim Datenaustausch zwischen z/OS-Systemen und IT-Systemen, die mit ASCII oder Unicode arbeiten (z. B. auch von USS nach z/OS), besteht die Gefahr, dass Informationen verfälscht werden, wenn fehlerhafte Übersetzungstabellen (Code Page Translation) eingesetzt werden. Besonders häufig betroffen ist dabei die Übersetzung von Sonderzeichen.

3. Anforderungen

Im Folgenden sind die spezifischen Anforderungen des Bausteins SYS.1.7 *IBM Z* aufgeführt. Der oder die Informationssicherheitsbeauftragte (ISB) ist dafür zuständig, dass alle Anforderungen gemäß dem festgelegten Sicherheitskonzept erfüllt und überprüft werden. Bei strategischen Entscheidungen ist der oder die ISB stets einzubeziehen.

Im IT-Grundschutz-Kompendium sind darüber hinaus weitere Rollen definiert. Sie sollten besetzt werden, insofern dies sinnvoll und angemessen ist.

Zuständigkeiten	Rollen
Grundsätzlich zuständig	IT-Betrieb
Weitere Zuständigkeiten	Vorgesetzte

Genau eine Rolle sollte *Grundsätzlich zuständig* sein. Darüber hinaus kann es noch *Weitere Zuständigkeiten* geben. Falls eine dieser weiteren Rollen für die Erfüllung einer Anforderung vorrangig zuständig ist, dann wird diese Rolle hinter der Überschrift der Anforderung in eckigen Klammern aufgeführt. Die Verwendung des Singulars oder Plurals sagt nichts darüber aus, wie viele Personen diese Rollen ausfüllen sollen.

3.1. Basis-Anforderungen

Die folgenden Anforderungen MÜSSEN für diesen Baustein vorrangig erfüllt werden.

SYS.1.7.A1 Einsatz restriktiver z/OS-Kennungen (B)

Berechtigungen mit hoher Autorisierung DÜRFEN NUR an Benutzende vergeben werden, die diese Rechte für ihre Tätigkeiten benötigen. Insbesondere die RACF-Attribute SPECIAL, OPERATIONS, AUDITOR und die entsprechenden GROUP-Attribute sowie die User-ID 0 unter den Unix System Services (USS) MÜSSEN restriktiv gehandhabt werden. Die Vergabe und der Einsatz dieser Berechtigungen MÜSSEN nachvollziehbar sein. Die besondere Kennung IBMUSER DARF NUR bei der Neuinstallation zur Erzeugung von Kennungen mit Attribut SPECIAL benutzt werden. Diese Kennung MUSS danach dauerhaft gesperrt werden. Um zu vermeiden, dass Administrierende sich dauerhaft aussperren, MUSS ein Notfall-User-Verfahren eingerichtet werden.

SYS.1.7.A2 Absicherung sicherheitskritischer z/OS-Dienstprogramme (B)

Sicherheitskritische (Dienst-)Programme und Kommandos sowie deren Alias-Namen MÜSSEN mit Rechten auf entsprechende RACF-Profile so geschützt werden, dass sie nur von den dafür vorgesehenen und autorisierten Personen benutzt werden können. Es MUSS sichergestellt sein, dass (Fremd-)Programme nicht unerlaubt installiert werden können. Außerdem DÜRFEN Programme NUR von gesicherten Quellen und über nachvollziehbare Methoden (z. B. SMP/E) installiert werden.

SYS.1.7.A3 Wartung von Z-Systemen (B)

Die Z-Hardware und -Firmware, das Betriebssystem sowie die verschiedenen Programme MÜSSEN regelmäßig und bei Bedarf gepflegt werden. Die hierfür notwendigen Wartungsaktivitäten MÜSSEN geplant und in das Änderungsmanagement (siehe OPS.1.1.3 *Patch- und Änderungsmanagement*) integriert werden. Insbesondere DÜRFEN Updates durch das herstellende Unternehmen NUR unter Kontrolle der Betreibenden durchgeführt werden, lokal über SE (Support Elements) bzw. HMC (Hardware Management Console) oder remote über die RSF (Remote Support Facility).

SYS.1.7.A4 Schulung des z/OS-Bedienungspersonals (B) [Vorgesetzte]

Administrierende, Bedienende und Prüfende im z/OS-Bereich MÜSSEN entsprechend ihren Aufgaben ausgebildet sein. Insbesondere MÜSSEN RACF-Administrierende mit dem Sicherheitssystem selbst sowie gegebenenfalls mit den weiteren für sie relevanten Funktionen vertraut sein.

SYS.1.7.A5 Einsatz und Sicherung systemnaher z/OS-Terminals (B)

Systemnahe z/OS-Terminals MÜSSEN physisch gegen unbefugten Zutritt und logisch gegen unbefugten Zugang geschützt werden. Insbesondere die Support-Elemente sowie die HMC-, MCS-, SMCS-, Extended MCS- und Monitor-Konsolen MÜSSEN dabei berücksichtigt werden. Voreingestellte Passwörter MÜSSEN geändert werden. Zugänge über Webserver und andere Fernzugänge MÜSSEN durch Verschlüsselung geschützt werden. Nicht benötigte Webserver und Fernzugänge MÜSSEN deaktiviert werden, wenn sie nicht benötigt werden.

SYS.1.7.A6 Einsatz und Sicherung der Remote Support Facility (B)

Es MUSS entschieden werden, ob und gegebenenfalls wie RSF eingesetzt wird. Der Einsatz MUSS im Wartungsvertrag vorgesehen und mit dem Hardware-Support abgestimmt sein. Es MUSS sichergestellt werden, dass die RSF-Konfiguration nur von hierzu autorisierten Personen geändert werden kann. Wartungszugriffe für Firmware-Modifikationen durch das herstellende Unternehmen MÜSSEN von Betreibenden explizit freigegeben und nach Beendi-

gung wieder deaktiviert werden. Die RSF-Kommunikation MUSS über Proxy-Server und zusätzlich über gesicherte Verbindungen (wie TLS) stattfinden.

SYS.1.7.A7 Restriktive Autorisierung unter z/OS (B)

Im Rahmen der Grundkonfiguration MÜSSEN die Autorisierungsmechanismen so konfiguriert werden, dass alle Personen (definierte User-IDs in Gruppen gemäß Rolle) nur die Zugriffsmöglichkeiten erhalten, die sie für ihre jeweiligen Tätigkeiten benötigen. Hierfür MÜSSEN insbesondere APF-Autorisierungen (Authorized Program Facility), SVCs (SuperVisor Calls), Ressourcen des z/OS-Betriebssystems, IPL-Parameter, Parmlib-Definitionen, Started Tasks und JES2/3-Definitionen berücksichtigt werden.

SYS.1.7.A8 Einsatz des z/OS-Sicherheitssystems RACF (B)

Der Einsatz von RACF für z/OS MUSS sorgfältig geplant werden, dazu gehören auch die Auswahl des Zeichensatzes, die Festlegung von Regeln für User-ID und Passwort sowie die Aktivierung der KDFAES-Verschlüsselung. Falls RACF PassTickets verwendet werden, MUSS der Enhanced PassTicket Algorithmus aktiviert werden. Voreingestellte Passwörter für das RVARY-Kommando und für neu angelegte User-IDs MÜSSEN geändert werden. Falls RACF-Exits eingesetzt werden sollen, MUSS deren Einsatz begründet, dokumentiert und regelmäßig überwacht werden.

Für das Anlegen, Sperren, Freischalten und Löschen von RACF-Kennungen MÜSSEN geeignete Vorgehensweisen festgelegt sein. Nach einer festgelegten Anzahl fehlgeschlagener Anmeldeversuche MUSS eine RACF-Kennung gesperrt werden (Ausnahme: Notfall-User-Verfahren). Kennungen von Benutzenden MÜSSEN außerdem nach längerer Inaktivität gesperrt werden, Kennungen von Verfahren hingegen nicht.

Dateien, Started Tasks und sicherheitskritische Programme MÜSSEN mittels RACF-Profilen geschützt werden. Benutzende DÜRFEN darüber NUR die Zugriffsmöglichkeiten erhalten, die sie gemäß ihrer Rolle benötigen. Es MUSS außerdem sichergestellt sein, dass Benutzende ihre Zugriffsmöglichkeiten nicht unerlaubt erweitern können.

SYS.1.7.A9 Mandantenfähigkeit unter z/OS (B)

Falls ein z/OS-System von Mandanten genutzt werden soll, MUSS ein RACF-Konzept zur Mandantentrennung erstellt werden. Die Daten und Anwendungen der Mandanten MÜSSEN durch RACF-Profile getrennt werden. Hohe Berechtigungen im RACF (SPECIAL, OPERATIONS, AUDITOR) und ändernden Zugriff auf Dateien des z/OS-Betriebssystems DÜRFEN NUR Mitarbeitende der Betreibenden haben. Die Wartungsfenster, in denen das z/OS-System nicht zur Verfügung steht, MÜSSEN mit allen Mandanten, die auf dem betroffenen System arbeiten, abgestimmt werden.

SYS.1.7.A10 ENTFALLEN (B)

Diese Anforderung ist entfallen.

SYS.1.7.A11 Schutz der Session-Daten (B)

Session-Daten für die Verbindungen der RACF-Administrierenden und möglichst auch der anderen Mitarbeitenden MÜSSEN verschlüsselt übertragen werden.

3.2. Standard-Anforderungen

Gemeinsam mit den Basis-Anforderungen entsprechen die folgenden Anforderungen dem Stand der Technik für diesen Baustein. Sie SOLLTEN grundsätzlich erfüllt werden.

SYS.1.7.A12 ENTFALLEN (S)

Diese Anforderung ist entfallen.

SYS.1.7.A13 ENTFALLEN (S)

Diese Anforderung ist entfallen.

SYS.1.7.A14 Berichtswesen zum sicheren Betrieb von z/OS (S)

Zur Überwachung aller sicherheitsrelevanten Tätigkeiten unter z/OS SOLLTE ein Prozess eingerichtet werden. In diesem SOLLTE festgelegt sein, welche Sicherheitsreports regelmäßig erstellt werden, welche Tools und Datenquellen dabei herangezogen werden (z. B. System Management Facility) und wie mit Abweichungen von den Vorgaben umgegangen wird. Die Sicherheitsreports SOLLTEN bei Überprüfungen als Information verwendet werden.

SYS.1.7.A15 ENTFALLEN (S)

Diese Anforderung ist entfallen.

SYS.1.7.A16 Überwachung von z/OS-Systemen (S)

Während des Betriebs SOLLTE das z/OS-System auf wichtige Meldungen, Ereignisse und die Einhaltung von Grenzwerten überwacht werden. Insbesondere Fehlermeldungen auf der HMC-Konsole, WTOR- und wichtige WTO-Nachrichten (Write To Operator/with Reply), System Tasks, Sicherheitsverletzungen, Kapazitätsgrenzen sowie die Systemauslastung SOLLTEN berücksichtigt werden. Für die Überwachung SOLLTEN außerdem mindestens die MCS-Konsole, die System Management Facility, das SYSLOG und die relevanten Protokolldaten der Anwendungen herangezogen werden. Es SOLLTE gewährleistet sein, dass alle wichtigen Meldungen zeitnah erkannt werden und, falls notwendig, in geeigneter Weise darauf reagiert wird. Systemnachrichten SOLLTEN dabei so gefiltert werden, dass nur die wichtigen Nachrichten dargestellt werden.

SYS.1.7.A17 Synchronisierung von z/OS-Passwörtern und RACF-Kommandos (S)

Falls z/OS-Passwörter oder RACF-Kommandos über mehrere z/OS-Systeme automatisch synchronisiert werden sollen, SOLLTEN die jeweiligen Systeme möglichst weitgehend standardisiert sein. Die Sperrung von Kennungen durch Fehleingaben von Passwörtern SOLLTE NICHT synchronisiert werden. Das Risiko durch Synchronisation sicherheitskritischer RACF-Kommandos SOLLTE berücksichtigt werden. Die Verwaltungsfunktion des Synchronisationsprogramms SOLLTE nur autorisierten Mitarbeitenden im Rahmen ihrer Tätigkeit zur Verfügung stehen.

SYS.1.7.A18 Rollenkonzept für z/OS-Systeme (S)

Mindestens für die Systemverwaltung von z/OS-Systemen SOLLTE ein Rollenkonzept eingeführt werden. Für alle wichtigen Rollen der Systemverwaltung SOLLTEN außerdem Stellvertretungsregelungen vorhanden sein. Die RACF-Attribute SPECIAL, OPERATIONS und AUDITOR SOLLTEN verschiedenen Personen zugeordnet werden (Rollentrennung).

SYS.1.7.A19 Absicherung von z/OS-Transaktionsmonitoren (S)

Falls Transaktionsmonitore oder Datenbanken unter z/OS eingesetzt werden, wie beispielsweise IMS, CICS oder Db2, SOLLTEN diese mittels RACF abgesichert werden. Dies gilt auch für die zugehörigen System-Kommandos und -Dateien. Interne Sicherheitsmechanismen der Transaktionsmonitore und Datenbanken SOLLTEN hingegen nur dort angewandt werden, wo es keine entsprechenden RACF-Funktionen gibt. Benutzende und Administrierende SOLLTEN nur die Zugriffsrechte erhalten, die sie für ihre jeweilige Tätigkeit benötigen.

SYS.1.7.A20 Stilllegung von z/OS-Systemen (S)

Bei der Stilllegung von z/OS-Systemen SOLLTEN andere z/OS-Systeme, Verbünde und Verwaltungssysteme so angepasst werden, dass sie nicht mehr auf das stillgelegte System verweisen. Auch die Auswirkungen auf die Software-Lizenzen SOLLTEN berücksichtigt werden.

Datenträger, die vertrauliche Daten enthalten, SOLLTEN so gelöscht werden, dass ihr Inhalt nicht mehr reproduziert werden kann. Für den Fall, dass defekte Datenträger durch das herstellende Unternehmen ausgetauscht werden, SOLLTE vertraglich vereinbart sein, dass diese Festplatten sicher vernichtet oder so gelöscht werden, dass ihr Inhalt nicht mehr reproduziert werden kann.

SYS.1.7.A21 Absicherung des Startvorgangs von z/OS-Systemen (S)

Die für den Startvorgang eines z/OS-Systems notwendigen Parameter SOLLTEN dokumentiert und dem Operating-Personal bekannt sein. Außerdem SOLLTEN die erforderlichen Hardware-Konfigurationen vorliegen, wie die IOCDS-Datei (Input/Output Configuration Data Set) und die LPARs (Logical Partitions). Eine z/OS-Master-Konsole und eine Backup-Konsole SOLLTEN definiert sein, um Nachrichten kontrollieren zu können. Nach dem Startvorgang SOLLTE

anhand einer Prüfliste kontrolliert werden, ob der Systemstatus den Soll-Vorgaben entspricht. Darüber hinaus SOLLTE eine Fallback-Konfiguration vorgehalten werden, mit der das System vor der letzten Änderung erfolgreich gestartet worden ist.

SYS.1.7.A22 Absicherung der Betriebsfunktionen von z/OS (S)

Alle die Produktion beeinflussenden Wartungsarbeiten sowie dynamische und sonstige Änderungen SOLLTEN nur im Rahmen des Änderungsmanagements durchgeführt werden (siehe OPS.1.1.3 *Patch- und Änderungsmanagement*). SDSF (System Display and Search Facility) und ähnliche Funktionen sowie die Prioritäten-Steuerung für Jobs SOLLTEN mittels RACF vor unberechtigtem Zugriff geschützt werden. z/OS-System-Kommandos und besonders sicherheitsrelevante Befehle für dynamische Änderungen SOLLTEN über RACF geschützt werden. Bei der dynamischen Definition von Hardware SOLLTE sichergestellt werden, dass eine Ressource im Wirkbetrieb nicht mehreren Einzelsystemen außerhalb eines Parallel Sysplex zugeordnet wird.

SYS.1.7.A23 Absicherung von z/VM (S)

Falls z/VM eingesetzt wird, SOLLTE das Produkt in das Patch-Management integriert werden. Alle voreingestellten Passwörter SOLLTEN geändert werden. Die Rolle des z/VM-Systemadministrierenden SOLLTE nur an Personen vergeben werden, die diese Berechtigungen benötigen. Die Sicherheitsadministration von z/VM SOLLTE über RACF für z/VM erfolgen. Passwörter von realen *Usern* und *Guest-Usern* SOLLTEN mittels RACF für z/VM verschlüsselt werden. Die sicherheitskritischen Systemkommandos von z/VM SOLLTEN über RACF geschützt werden. Unter z/VM definierte virtuelle Maschinen SOLLTEN nur die für die jeweiligen Aufgaben notwendigen Ressourcen erhalten und strikt voneinander getrennt sein. Unter z/VM SOLLTEN nur die benötigten Dienste gestartet werden. Wenn Überprüfungen durchgeführt werden, SOLLTEN die Journaling-Funktion von z/VM und die Audit-Funktionen von RACF eingesetzt werden.

SYS.1.7.A24 Datenträgerverwaltung unter z/OS-Systemen (S)

Dateien, Programme und Funktionen zur Verwaltung von Datenträgern sowie die Datenträger selbst (Festplatten und Bänder) einschließlich Master-Katalog SOLLTEN mittels RACF-Profilen geschützt werden. Es SOLLTEN Sicherungskopien aller wichtigen Dateien zur Verfügung stehen, die in einer Notfallsituation zurückgespielt werden können. Die Zuordnung von Datenträgern zu den Z-Systemen SOLLTE nachvollziehbar sein. Es SOLLTE gewährleistet werden, dass je nach Volumen und Zeitfenster genügend Bandstationen zur Verfügung stehen. Beim Einsatz des HSM (Hierarchical Storage Manager) SOLLTE festgelegt werden, welche Festplatten gesichert werden sollen und wie die Sicherung erfolgen soll. Bänder, die vom HSM verwaltet werden, SOLLTEN NICHT anderweitig bearbeitet werden.

SYS.1.7.A25 Festlegung der Systemdimensionierung von z/OS (S)

Die Grenzen für die maximale Belastung der Ressourcen (Anzahl der CPUs, Speicher, Bandbreite etc.) SOLLTEN den Hardware-Voraussetzungen entsprechend festgelegt und den zuständigen Administrierenden und Anwendungszuständigen bekannt sein. Die Anzahl der zur Verfügung stehenden Magnetband-Stationen, die Zeiten, zu denen Anwendungen auf Magnetband-Stationen zugreifen und die benötigten Festplattenkapazitäten SOLLTEN mit den Anwendungszuständigen abgestimmt sein. Die Festplattenkapazitäten SOLLTEN außerdem vom Space-Management überwacht werden.

SYS.1.7.A26 WorkLoad Management für z/OS-Systeme (S)

Die Programme, Dateien und Kommandos des WorkLoad Managers (WLM) sowie die dafür notwendigen Couple Data Sets SOLLTEN mittels RACF geschützt werden. Dabei SOLLTE sichergestellt sein, dass die Berechtigungen zum Ändern des WLM über z/OS-Kommandos und über die SDSF-Schnittstelle gleich sind.

SYS.1.7.A27 Zeichensatzkonvertierung bei z/OS-Systemen (S)

Wenn Textdateien zwischen z/OS-Systemen und anderen Systemen übertragen werden, SOLLTE darauf geachtet werden, dass eventuell eine Zeichensatzkonvertierung erforderlich ist. Dabei SOLLTE die korrekte Umsetzungstabelle verwendet werden. Bei der Übertragung von Programm-Quellcode SOLLTE geprüft werden, ob alle Zeichen richtig übersetzt wurden. Bei der Übertragung von Binärdaten SOLLTE hingegen sichergestellt sein, dass keine Zeichensatzkonvertierung stattfindet.

SYS.1.7.A28 Lizenzschlüssel-Management für z/OS-Software (S)

Für Lizenzschlüssel, deren Gültigkeit zeitlich begrenzt ist, SOLLTE ein Verfahren zur rechtzeitigen Erneuerung eingerichtet sein. Die Laufzeiten der Lizenzschlüssel SOLLTEN dokumentiert werden. Die Dokumentation SOLLTE allen betroffenen Administrierenden zur Verfügung stehen.

SYS.1.7.A29 Absicherung von Unix System Services bei z/OS-Systemen (S)

Die Parameter der Unix System Services (USS) SOLLTEN entsprechend der funktionalen und sicherheitstechnischen Vorgaben sowie unter Berücksichtigung der verfügbaren Ressourcen eingestellt werden. HFS- und zFS-Dateien, die USS-Dateisysteme enthalten, SOLLTEN über RACF-Profile abgesichert und in das Datensicherungskonzept einbezogen werden. Außerdem SOLLTE das Root-Dateisystem mit der Option Read-Only gemounted werden. Es SOLLTE im USS-Dateisystem KEINE APF-Autorisierung (Authorized Program Facility) über das File Security Packet (FSP) geben. Stattdessen SOLLTEN die Module von APF-Dateien des z/OS-Betriebssystems geladen werden. Zwischen USS-User-IDs und z/OS-User-IDs SOLLTE es eine eindeutige Zuordnung geben. Berechtigungen unter USS SOLLTEN über die RACF-Klasse UNIXPRIV vergeben werden und NICHT durch Vergabe der UID 0. Für Überprüfungen und das Monitoring der USS SOLLTEN die gleichen Mechanismen wie für z/OS genutzt werden.

SYS.1.7.A30 Absicherung der z/OS-Trace-Funktionen (S)

Die Trace-Funktionen von z/OS wie GTF (Generalized Trace Facility), NetView oder ACF/TAP (Advanced Communication Function/Trace Analysis Program) und die entsprechenden Dateien SOLLTEN so geschützt werden, dass nur die zuständigen und autorisierten Mitarbeitenden darauf Zugriff haben. Die Trace-Funktion von NetView SOLLTE deaktiviert sein und nur im Bedarfsfall aktiviert werden.

SYS.1.7.A31 Notfallvorsorge für z/OS-Systeme (S)

Es SOLLTE ein Verfahren zur Wiederherstellung einer funktionierenden RACF-Datenbank vorgesehen sein. Weiterhin SOLLTEN eine Kopie des z/OS-Betriebssystems als z/OS-Backup-System und, unabhängig von Produktivsystem, ein z/OS-Notfallsystem vorgehalten werden.

3.3. Anforderungen bei erhöhtem Schutzbedarf

Im Folgenden sind für diesen Baustein exemplarische Vorschläge für Anforderungen aufgeführt, die über dasjenige Schutzniveau hinausgehen, das dem Stand der Technik entspricht. Die Vorschläge SOLLTEN bei erhöhtem Schutzbedarf in Betracht gezogen werden. Die konkrete Festlegung erfolgt im Rahmen einer individuellen Risikoanalyse.

SYS.1.7.A32 Festlegung von Standards für z/OS-Systemdefinitionen (H)

Es SOLLTEN Standards und Namenskonventionen für z/OS-Systemdefinitionen festgelegt und dokumentiert werden. Die Dokumentationen SOLLTEN den Administrierenden zur Verfügung stehen. Die Einhaltung der Standards SOLLTE regelmäßig überprüft werden. Standards SOLLTEN insbesondere für Datei-, Datenbank-, Job- und Volume-Namen sowie für Application-, System- und User-IDs definiert werden.

SYS.1.7.A33 Trennung von Test- und Produktionssystemen unter z/OS (H)

Es SOLLTEN technische Maßnahmen ergriffen werden, um Entwicklungs- und Testsysteme von Produktionssystemen unter z/OS zu trennen. Dabei SOLLTEN eventuelle Zugriffsmöglichkeiten über gemeinsame Festplatten und den Parallel Sysplex beachtet werden.

SYS.1.7.A34 Batch-Job-Planung für z/OS-Systeme (H)

Wenn ein z/OS-System eine größere Anzahl von Batch-Jobs verarbeitet, SOLLTE für die Ablaufsteuerung der Batch-Jobs ein Job-Scheduler eingesetzt werden. Der Job-Scheduler sowie die zugehörigen Dateien und Tools SOLLTEN mittels RACF geeignet geschützt werden.

SYS.1.7.A35 Einsatz von RACF-Exits (H)

Falls RACF-Exits eingesetzt werden, SOLLTEN die sicherheitstechnischen und betrieblichen Auswirkungen analysiert werden. Die RACF-Exits SOLLTEN außerdem über das SMP/E (System Modification Program/Enhanced) als USERMOD installiert und überwacht werden.

SYS.1.7.A36 Interne Kommunikation von Betriebssystemen (H)

Die Kommunikation von Betriebssystemen, z/OS oder Linux, die entweder im LPAR-Mode oder unter z/VM auf derselben Z-Hardware installiert sind, SOLLTE über interne Kanäle erfolgen, d. h. über HiperSockets oder virtuelle CTC-Verbindungen (Channel-to-Channel).

SYS.1.7.A37 Parallel Sysplex unter z/OS (H)

Anhand der Verfügbarkeits- und Skalierbarkeitsanforderungen SOLLTE entschieden werden, ob ein Parallel Sysplex (Cluster von z/OS-Systemen) eingesetzt wird und gegebenenfalls welche Redundanzen dabei vorgesehen werden. Bei der Dimensionierung der Ressourcen SOLLTEN die Anforderungen der Anwendungen berücksichtigt werden. Die Software und die Definitionen der LPARs des Sysplex, einschließlich RACF, SOLLTEN synchronisiert oder als gemeinsam benutzte Dateien bereitgestellt sein.

Es SOLLTE sichergestellt sein, dass alle LPARs des Sysplex auf die Couple Data Sets zugreifen können. Die Couple Data Sets sowie alle sicherheitskritischen Programme und Kommandos zur Verwaltung des Sysplex SOLLTEN mittels RACF geschützt werden. Außerdem SOLLTE ein GRS-Verbund (Global Resource Serialization) eingerichtet werden. Die Festplatten des Sysplexes SOLLTEN strikt von den Festplatten anderer Systeme getrennt werden. Der System Logger SOLLTE mit Staging Data Set eingesetzt werden.

SYS.1.7.A38 Einsatz des VTAM Session Management Exit unter z/OS (H)

Falls ein VTAM Session Management Exit eingesetzt werden soll, SOLLTE gewährleistet werden, dass dadurch der sichere und performante Betrieb nicht beeinträchtigt wird. Der Exit SOLLTE mindestens eine nachträgliche Kontrolle der abgewiesenen Login-Versuche ermöglichen. Außerdem SOLLTE sich der Exit dynamisch konfigurieren lassen und das Regelwerk von einer externen Datei nachladen. Funktionen, Kommandos und Dateien im Zusammenhang mit dem Exit SOLLTEN durch RACF geschützt werden.

4. Weiterführende Informationen

4.1. Wissenswertes

Im Umfeld von Z-Systemen sind eine Reihe von Abkürzungen gebräuchlich, die nicht an anderen Stellen im IT-Grundschutz erläutert werden. Hierzu gehören:

- HMC (Hardware Management Console), MCS (Multiple Console Support), SMCS, Extended MCS: Konsolen zur Steuerung und Kontrolle eines Z-Systems bzw. z/OS-Betriebssystems
- HFS: Hierarchical File System, Hierarchisches Dateisystem
- IPL: Initial Program Load, Startvorgang eines Betriebssystems
- RSF: Remote Support Facility
- SE: Support Elements, zur Konfiguration und Kontrolle des Systems
- SMP/E: System Modification Program/Extended, Verfahren zur Software-Installation
- zFS: zSeries File System, Dateisystem, das unter z/OS und Unix System Services (USS) eingesetzt wird.

Der Hersteller IBM gibt weitere Informationen zum Thema IBM Z in „ABC of z/OS System Programming Volume 1-13“, IBM Redbooks, https://www.redbooks.ibm.com.

Bundesamt
für Sicherheit in der
Informationstechnik

SYS.1.8 Speicherlösungen

1. Beschreibung

1.1. Einleitung

Das stetige Wachstum digitaler Informationen und die zunehmende Menge unstrukturierter Informationen führen dazu, dass innerhalb von Institutionen zentrale Speicherlösungen eingesetzt werden. Dabei unterliegen die Anforderungen an solche Speicherlösungen einem stetigen Wandel, der sich beispielsweise an folgenden Aspekten beobachten lässt:

- Die Daten einer Institution sollen jederzeit, an jedem Ort und für unterschiedliche Anwendungsszenarien verfügbar sein. Dadurch gelten für moderne Speicherlösungen häufig gestiegene Verfügbarkeitsanforderungen.
- Die zunehmende Digitalisierung sämtlicher Informationen in einer Institution macht es notwendig, dass weitreichende rechtliche Vorgaben (Compliance-Anforderungen) beachtet und eingehalten werden müssen.
- Speicherlösungen sollen dynamisch an die sich stetig ändernden Anforderungen anpassbar sein und Speicherplatz zentral bereitstellen können.

In der Vergangenheit wurden Speicherlösungen oft umgesetzt, indem Speichermedien direkt an einen Server angeschlossen wurden. Diese sogenannten Direct-Attached-Storage-(DAS)-Systeme können die aktuellen und zukünftigen Anforderungen jedoch oft nicht mehr erfüllen. Daher sind die heute weitverbreiteten zentralen Speicherlösungen und deren Bestandteile notwendig, die sich wie folgt unterscheiden lassen:

- Speicherlösungen: Eine Speicherlösung besteht aus einem oder mehreren Speichernetzen sowie mindestens einem Speichersystem.
- Speichernetze: Speichernetze ermöglichen einerseits den Zugriff auf die Speichersysteme, andererseits die Replikation von Daten zwischen Speichersystemen.
- Speichersysteme: Als Speichersystem wird die zentrale Instanz bezeichnet, die für andere IT-Systeme Speicherplatz zur Verfügung stellt. Ein Speichersystem erlaubt außerdem den zeitgleichen Zugriff mehrerer IT-Systeme auf den vorhandenen Speicherplatz.

1.2. Zielsetzung

Ziel dieses Bausteins ist es, aufzuzeigen, wie zentrale Speicherlösungen sicher geplant, umgesetzt, betrieben und ausgesondert werden.

1.3. Abgrenzung und Modellierung

Der Baustein SYS.1.8 *Speicherlösungen* ist immer dann anzuwenden, wenn zentrale Speicherlösungen eingesetzt werden. Somit kann er auf Network-Attached-Storage-(NAS)-Systeme, Storage-Area-Networks-(SAN)-Systeme, Hybrid Storage, Objekt Storage oder Cloud Storage angewendet werden. Dabei muss jedoch Folgendes beachtet werden:

- **Network Attached Storage (NAS)** stellt beispielsweise über die Protokolle NFS (Network File System), AFP (Apple Filing Protocol) und CIFS (Common Internet File System) Zugriffe auf die Speichersysteme zur Verfügung. Der Hauptanwendungsfall besteht darin, Fileserver-Dienste zur Verfügung zu stellen. Für NAS-Systeme sind daher auch zusätzlich zu diesem Baustein die Bausteine SYS.1.1 *Allgemeiner Server* sowie APP.3.3 *Fileserver* anzuwenden.

- **Storage Area Networks (SAN)** werden in der Regel durch ein dediziertes Speichernetz zwischen Speichersystemen und angeschlossenen IT-Systemen geschaffen. Für SAN-Systeme ist daher der Baustein NET.1.1 *Netzarchitektur und -design* geeignet zu berücksichtigen. Speichersysteme, die sowohl über NAS als auch SAN Daten zur Verfügung stellen können, werden oft unter der Bezeichnung **Hybrid-Storage** oder kombiniertes Speichersystem (Unified Storage) geführt. Für Hybrid-Systeme sind daher auch zusätzlich die Bausteine SYS.1.1 *Allgemeiner Server* sowie APP.3.3 *Fileserver* anzuwenden. Darüber hinaus ist der Baustein NET.1.1 *Netzarchitektur und -design* geeignet zu berücksichtigen.
- **Objekt-Storage** (oftmals auch als **Object-based Storage** bezeichnet) ermöglicht gegenüber den traditionellen blockbasierten und dateibasierten Zugriffsmethoden einen objektbasierten Zugriff auf Daten. Der Zugriff auf einen objektbasierenden Speicher erfolgt über eine führende Anwendung. Die Anwendung greift hierbei über eine spezielle Schnittstelle (Application Programming Interface (API)) und deren mögliche Kommandos oder direkt per Internet Protocol (IP) auf den Objekt-Storage zu. Für objektbasierende Speicherlösungen ist daher auch zusätzlich der Baustein SYS.1.1 *Allgemeiner Server* anzuwenden. Darüber hinaus müssen Sicherheitsanforderungen mitberücksichtigt werden, die sich dadurch ergeben, dass Webservices eingesetzt werden. Webservices werden im vorliegenden Baustein nicht betrachtet.
- Im Zusammenhang mit Weiterentwicklungen im Speicherumfeld etabliert sich zunehmend auch der Begriff des **Cloud Storage**. Hierunter sind Speicherlösungen als Basis für Cloud-Services zu verstehen. Die Speicherlösung an sich bleibt dabei weitgehend unverändert, jedoch liegt eine von den klassischen SAN- oder NAS-Architekturen abweichende Art des Zugriffs auf die gespeicherten Daten vor. Dieser wird in der Regel mittels Web-Service-Schnittstelle (via Representional State Transfer (REST) und Simple Object Access Protocol (SOAP)) realisiert.

Datensicherungsgeräte, die an das Speichersystem oder an das Speichernetz angeschlossen sind, werden hier nicht betrachtet, sondern im Baustein OPS.1.2.2 *Archivierung* behandelt. Konzeptionelle Aspekte der Datensicherung werden im Baustein CON.3 *Datensicherungskonzept* erläutert.

Oft kann eine Vielzahl von Konten auf Speicherlösungen zugreifen. Deswegen sollten Speicherlösungen geeignet im Rollen- und Rechtekonzept mit berücksichtigt werden. Anforderungen dazu finden sich im Baustein ORP.4 *Identitäts- und Berechtigungsmanagement*.

Falls auf externe Dienstleistung zurückgegriffen wird, um eine Speicherlösung zu betreiben, müssen die Anforderungen des Bausteins OPS.2.3 *Nutzung von Outsourcing* gesondert berücksichtigt werden.

2. Gefährdungslage

Da IT-Grundschutz-Bausteine nicht auf individuelle Informationsverbünde eingehen können, werden zur Darstellung der Gefährdungslage typische Szenarien zugrunde gelegt. Die folgenden spezifischen Bedrohungen und Schwachstellen sind für den Baustein SYS.1.8 *Speicherlösungen* von besonderer Bedeutung.

2.1. Unsichere Default-Einstellungen bei Speicherkomponenten

Häufig werden Speicherkomponenten mit einer Default-Konfiguration ausgeliefert, damit die Geräte schnell und mit möglichst umfassenden Funktionen in Betrieb genommen werden können. So sind in vielen Geräten nicht benötigte Protokolle aktiviert, wie z. B. HTTP, Telnet und unsichere SNMP-Versionen. Werden Speicherkomponenten mit unsicheren Werkseinstellungen produktiv eingesetzt, kann einfacher unberechtigt auf sie zugegriffen werden. Das kann dazu führen, dass z. B. Dienste nicht mehr verfügbar sind oder dass unerlaubt auf vertrauliche Informationen der Institution zugegriffen wird.

2.2. Manipulation von Daten über das Speichersystem

Über ein mangelhaft konfiguriertes Storage Area Network (SAN) können sich ungewollt Netze verbinden. Ist beispielsweise ein Server mit SAN-Anschluss aus dem Internet erreichbar und so von außen kompromittierbar, kann dieser als Einstiegspunkt genutzt werden, um unberechtigt auf schützenswerte Informationen zuzugreifen, die im SAN gespeichert sind. Da auf diese Weise alle Sicherheits- und Überwachungsmaßnahmen, wie Firewalls oder Intrusion Detection Systeme (IDS), in den Netzen einer Institution umgangen werden können, ist das Schadenspotenzial groß.

2.3. Verlust der Vertraulichkeit durch storagebasierte Replikationsmethoden

Storagebasierte Replikationsmethoden haben den Zweck, gespeicherte oder archivierte Daten in Echtzeit über ein Speichernetz zu duplizieren und diese damit zusätzlich redundant abzuspeichern. Hierdurch sollen Datenverluste vermieden werden. Die automatisierte Replikation unverschlüsselter Daten birgt allerdings sowohl im eigenen Netz als auch bei der Nutzung öffentlicher Netze Risiken. So kann unberechtigt auf Replikationsverkehr zugegriffen werden, beispielsweise mittels FC-Analysern (FC-Replikation) oder Sniffern (IP-Replikation).

2.4. Zugriff auf Informationen anderer Mandanten durch WWN-Spoofing

Geräte in einem FC-SAN werden intern über World Wide Names (WWNs) verwaltet und zugeordnet. Sie entsprechen in gewisser Weise den MAC-Adressen von Ethernet-Netzadaptern. Mittels Programmen, die durch das herstellende Unternehmen der Host Bus Adapter (HBA) zur Verfügung gestellt werden, kann der WWN eines HBAs geändert werden. Dadurch kann unberechtigt auf Daten zugegriffen werden. Die Manipulation von WWNs, auch als WWN-Spoofing bezeichnet, birgt für eine Institution erhebliches Gefahrenpotenzial. Insbesondere im Zusammenhang mit mandantenfähigen Speichersystemen können Unberechtigte auf die Informationen anderer Mandanten zugreifen.

2.5. Überwindung der logischen Netzseparierung

Werden die Netzstrukturen unterschiedlicher Mandanten nicht durch physisch getrennte Netze, sondern durch virtuelle Storage Area Networks (VSANs) separiert, kann hierdurch die Informationssicherheit der Institution gefährdet werden. Gelingt es Angreifenden, in das Netz eines anderen Mandanten einzudringen, können sie sowohl auf das virtuelle SAN dieses Mandanten als auch auf die übertragenen Nutzdaten zugreifen.

2.6. Ausfall von Komponenten einer Speicherlösung

Komplexe netzbasierte Speicherlösungen bestehen oft aus vielen Komponenten (z. B. FC-Switches, Storage Controller, Virtualisierungs-Appliance). Fallen einzelne Komponenten einer Speicherlösung aus, kann dies dazu führen, dass wichtige Anwendungen nicht mehr korrekt arbeiten und somit Datenverluste drohen.

2.7. Erlangung physischen Zugangs auf SAN-Switches

Existieren in einer Institution unzureichende Zutritts- und Zugangskontrollen zu den Komponenten eines Speichersystems oder fehlen diese gänzlich, kann es gelingen, sich physischen Zugang zu vorhandenen Switches zu verschaffen bzw. zusätzliche FC-SAN-Switches an das Netz anzuschließen. Ziel bei einem solchen Angriff könnte es sein, auf die verteilte Zoning-Datenbank zuzugreifen, um diese so zu verändern, dass auf die Speichersysteme zugegriffen werden kann.

3. Anforderungen

Im Folgenden sind die spezifischen Anforderungen des Bausteins SYS.1.8 *Speicherlösungen* aufgeführt. Der oder die Informationssicherheitsbeauftragte (ISB) ist dafür zuständig, dass alle Anforderungen gemäß dem festgelegten Sicherheitskonzept erfüllt und überprüft werden. Bei strategischen Entscheidungen ist der oder die ISB stets einzubeziehen.

Im IT-Grundschutz-Kompendium sind darüber hinaus weitere Rollen definiert. Sie sollten besetzt werden, insofern dies sinnvoll und angemessen ist.

Zuständigkeiten	Rollen
Grundsätzlich zuständig	IT-Betrieb
Weitere Zuständigkeiten	Haustechnik

Genau eine Rolle sollte *Grundsätzlich zuständig* sein. Darüber hinaus kann es noch *Weitere Zuständigkeiten* geben. Falls eine dieser weiteren Rollen für die Erfüllung einer Anforderung vorrangig zuständig ist, dann wird diese Rolle hinter der Überschrift der Anforderung in eckigen Klammern aufgeführt. Die Verwendung des Singulars oder Plurals sagt nichts darüber aus, wie viele Personen diese Rollen ausfüllen sollen.

3.1. Basis-Anforderungen

Die folgenden Anforderungen MÜSSEN für diesen Baustein vorrangig erfüllt werden.

SYS.1.8.A1 Geeignete Aufstellung von Speichersystemen (B) [Haustechnik]

Die IT-Komponenten von Speicherlösungen MÜSSEN in verschlossenen Räumen aufgestellt werden. Zu diesen Räumen DÜRFEN NUR Berechtigte Zutritt haben. Zudem MUSS eine sichere Stromversorgung sichergestellt sein. Die Vorgaben des herstellenden Unternehmens zur empfohlenen Umgebungstemperatur und Luftfeuchte MÜSSEN eingehalten werden.

SYS.1.8.A2 Sichere Grundkonfiguration von Speicherlösungen (B)

Bevor eine Speicherlösung produktiv eingesetzt wird, MUSS sichergestellt sein, dass alle eingesetzten Softwarekomponenten und die Firmware aktuell sind. Danach MUSS eine sichere Grundkonfiguration hergestellt werden.

Nicht genutzte Schnittstellen des Speichersystems MÜSSEN deaktiviert werden. Die Dateien zur Default-Konfiguration, zur vorgenommenen Grundkonfiguration und zur aktuellen Konfiguration SOLLTEN redundant und geschützt aufbewahrt werden.

SYS.1.8.A3 ENTFALLEN (B)

Diese Anforderung ist entfallen.

SYS.1.8.A4 Schutz der Administrationsschnittstellen (B)

Alle Administrations- und Management-Zugänge der Speichersysteme MÜSSEN eingeschränkt werden. Es MUSS sichergestellt sein, dass aus nicht-vertrauenswürdigen Netzen heraus nicht auf die Administrationsschnittstellen zugegriffen werden kann.

Es SOLLTEN als sicher geltende Protokolle eingesetzt werden. Sollten dennoch unsichere Protokolle verwendet werden, MUSS für die Administration ein eigenes Administrationsnetz (siehe NET.1.1 *Netzarchitektur und -design*) genutzt werden.

SYS.1.8.A5 ENTFALLEN (B)

Diese Anforderung ist entfallen.

3.2. Standard-Anforderungen

Gemeinsam mit den Basis-Anforderungen entsprechen die folgenden Anforderungen dem Stand der Technik für diesen Baustein. Sie SOLLTEN grundsätzlich erfüllt werden.

SYS.1.8.A6 Erstellung einer Sicherheitsrichtlinie für Speicherlösungen (S)

Ausgehend von der allgemeinen Sicherheitsrichtlinie der Institution SOLLTE eine spezifische Sicherheitsrichtlinie für Speicherlösungen erstellt werden. Darin SOLLTEN nachvollziehbar Vorgaben beschrieben sein, wie Speicherlösungen sicher geplant, administriert, installiert, konfiguriert und betrieben werden können.

Die Richtlinie SOLLTE allen für Speicherlösungen zuständigen Administrierenden bekannt und grundlegend für ihre Arbeit sein. Wird die Richtlinie verändert oder wird von den Vorgaben abgewichen, SOLLTE dies mit dem oder der ISB abgestimmt und dokumentiert werden. Es SOLLTE regelmäßig überprüft werden, ob die Richtlinie noch korrekt umgesetzt ist. Gegebenenfalls SOLLTE sie aktualisiert werden. Die Ergebnisse SOLLTEN sinnvoll dokumentiert werden.

SYS.1.8.A7 Planung von Speicherlösungen (S)

Bevor Speicherlösungen in einer Institution eingesetzt werden, SOLLTE eine Anforderungsanalyse durchgeführt werden. In der Anforderungsanalyse SOLLTEN unter anderem die Themen Performance und Kapazität betrachtet werden. Auf Basis der ermittelten Anforderungen SOLLTE dann eine detaillierte Planung für Speicherlösungen erstellt werden. Darin SOLLTEN folgende Punkte berücksichtigt werden:

- Auswahl von herstellenden Unternehmen und Liefernden,
- Entscheidung für oder gegen zentrale Verwaltungssysteme (Management-Systeme),

- Planung des Netzanschlusses,
- Planung der Infrastruktur sowie
- Integration in bestehende Prozesse.

SYS.1.8.A8 Auswahl einer geeigneten Speicherlösung (S)

Die technischen Grundlagen unterschiedlicher Speicherlösungen SOLLTEN detailliert beleuchtet werden. Die Auswirkungen dieser technischen Grundlagen auf den möglichen Einsatz in der Institution SOLLTEN geprüft werden. Die Möglichkeiten und Grenzen der verschiedenen Speichersystemarten SOLLTEN für die Verantwortlichen der Institution transparent dargestellt werden. Die Entscheidungskriterien für eine Speicherlösung SOLLTEN nachvollziehbar dokumentiert werden. Ebenso SOLLTE die Entscheidung für die Auswahl einer Speicherlösung nachvollziehbar dokumentiert werden.

SYS.1.8.A9 Auswahl von Liefernden für eine Speicherlösung (S)

Anhand der spezifizierten Anforderungen an eine Speicherlösung SOLLTEN geeignete Liefernde ausgewählt werden. Die Auswahlkriterien und die Entscheidung SOLLTEN nachvollziehbar dokumentiert werden. Außerdem SOLLTEN Aspekte der Wartung und Instandhaltung schriftlich in sogenannten Service-Level-Agreements (SLAs) festgehalten werden. Die SLAs SOLLTEN eindeutig und quantifizierbar sein. Es SOLLTE genau geregelt werden, wann der Vertrag mit den Liefernden endet.

SYS.1.8.A10 Erstellung und Pflege eines Betriebshandbuchs (S)

Es SOLLTE ein Betriebshandbuch erstellt werden. Darin SOLLTEN alle Regelungen, Anforderungen und Einstellungen dokumentiert werden, die erforderlich sind, um Speicherlösungen zu betreiben. Das Betriebshandbuch SOLLTE regelmäßig aktualisiert werden.

SYS.1.8.A11 Sicherer Betrieb einer Speicherlösung (S)

Das Speichersystem SOLLTE hinsichtlich der Verfügbarkeit der internen Anwendungen, der Systemauslastung sowie kritischer Ereignisse überwacht werden. Weiterhin SOLLTEN für Speicherlösungen feste Wartungsfenster definiert werden, in denen Änderungen durchgeführt werden können. Insbesondere Firmware- oder Betriebssystemupdates von Speichersystemen oder den Netzkomponenten einer Speicherlösung SOLLTEN ausschließlich innerhalb eines solchen Wartungsfensters durchgeführt werden.

SYS.1.8.A12 ENTFALLEN (S)

Diese Anforderung ist entfallen.

SYS.1.8.A13 Überwachung und Verwaltung von Speicherlösungen (S)

Speicherlösungen SOLLTEN überwacht werden. Dabei SOLLTEN alle erhobenen Daten (Nachrichten) vorrangig daraufhin geprüft werden, ob die Vorgaben des Betriebshandbuchs eingehalten werden.

Die wesentlichen Nachrichten SOLLTEN mit Nachrichtenfilter herausgefiltert werden. Einzelne Komponenten der Speicherlösung und des Gesamtsystems SOLLTEN zentral verwaltet werden.

SYS.1.8.A14 Absicherung eines SANs durch Segmentierung (S)

Ein SAN SOLLTE segmentiert werden. Es SOLLTE ein Konzept erarbeitet werden, das die SAN-Ressourcen den jeweiligen Servern zuordnet. Hierfür SOLLTE anhand der Sicherheitsanforderungen und des Administrationsaufwands entschieden werden, welche Segmentierung in welcher Implementierung (z. B. FC-SANs oder iSCSI-Speichernetze) eingesetzt werden soll. Die aktuelle Ressourcenzuordnung SOLLTE mithilfe von Verwaltungswerkzeugen einfach und übersichtlich erkennbar sein. Weiterhin SOLLTE die aktuelle Zoning-Konfiguration dokumentiert werden. Die Dokumentation SOLLTE auch in Notfällen verfügbar sein.

SYS.1.8.A15 Sichere Trennung von Mandanten in Speicherlösungen (S)

Es SOLLTE definiert und nachvollziehbar dokumentiert werden, welche Anforderungen die Institution an die Mandantenfähigkeit einer Speicherlösung stellt. Die eingesetzten Speicherlösungen SOLLTEN diese dokumentierten Anforderungen erfüllen.

Im Block-Storage-Umfeld SOLLTE *LUN Masking* eingesetzt werden, um Mandanten voneinander zu trennen. In Fileservice-Umgebungen SOLLTE es möglich sein, mit virtuellen Fileservern zu agieren. Dabei SOLLTE jedem Mandanten ein eigener Fileservice zugeordnet werden.

Beim Einsatz von IP oder iSCSI SOLLTEN die Mandanten über eine Segmentierung im Netz voneinander getrennt werden. Wird Fibre Channel eingesetzt, SOLLTE mithilfe von VSANs und Soft-Zoning separiert werden.

SYS.1.8.A16 Sicheres Löschen in SAN-Umgebungen (S)

In mandantenfähigen Speichersystemen SOLLTE sichergestellt werden, dass Logical Unit Numbers (LUNs), die einem bestimmten Mandanten zugeordnet sind, gelöscht werden.

SYS.1.8.A17 Dokumentation der Systemeinstellungen von Speichersystemen (S)

Alle Systemeinstellungen von Speichersystemen SOLLTEN dokumentiert werden. Die Dokumentation SOLLTE die technischen und organisatorischen Vorgaben sowie alle spezifischen Konfigurationen der Speichersysteme der Institution enthalten.

Sofern die Dokumentation der Systemeinstellungen vertrauliche Informationen beinhaltet, SOLLTEN diese vor unberechtigtem Zugriff geschützt werden. Die Dokumentation SOLLTE regelmäßig überprüft werden. Sie SOLLTE immer aktuell sein.

SYS.1.8.A18 Sicherheitsaudits und Berichtswesen bei Speichersystemen (S)

Alle eingesetzten Speichersysteme SOLLTEN regelmäßig auditiert werden. Dafür SOLLTE ein entsprechender Prozess eingerichtet werden. Es SOLLTE geregelt werden, welche Sicherheitsreports mit welchen Inhalten regelmäßig zu erstellen sind. Zudem SOLLTE auch geregelt werden, wie mit Abweichungen von Vorgaben umgegangen wird und wie oft und in welcher Tiefe Audits durchgeführt werden.

SYS.1.8.A19 Aussonderung von Speicherlösungen (S)

Werden ganze Speicherlösungen oder einzelne Komponenten einer Speicherlösung nicht mehr benötigt, SOLLTEN alle darauf vorhandenen Daten auf andere Speicherlösungen übertragen werden. Hierfür SOLLTE eine Übergangsphase eingeplant werden. Anschließend SOLLTEN alle Nutzdaten und Konfigurationsdaten sicher gelöscht werden. Aus allen relevanten Dokumenten SOLLTEN alle Verweise auf die außer Betrieb genommene Speicherlösung entfernt werden.

SYS.1.8.A20 Notfallvorsorge und Notfallreaktion für Speicherlösungen (S)

Es SOLLTE ein Notfallplan für die eingesetzte Speicherlösung erstellt werden. Der Notfallplan SOLLTE genau beschreiben, wie in bestimmten Notfallsituationen vorzugehen ist. Auch SOLLTEN Handlungsanweisungen in Form von Maßnahmen und Kommandos enthalten sein, die die Fehleranalyse und Fehlerkorrektur unterstützen. Um Fehler zu beheben, SOLLTEN geeignete Werkzeuge eingesetzt werden.

Regelmäßige Übungen und Tests SOLLTEN anhand des Notfallplans durchgeführt werden. Nach den Übungen und Tests sowie nach einem tatsächlichen Notfall SOLLTEN die dabei erzeugten Daten sicher gelöscht werden.

3.3. Anforderungen bei erhöhtem Schutzbedarf

Im Folgenden sind für diesen Baustein exemplarische Vorschläge für Anforderungen aufgeführt, die über dasjenige Schutzniveau hinausgehen, das dem Stand der Technik entspricht. Die Vorschläge SOLLTEN bei erhöhtem Schutzbedarf in Betracht gezogen werden. Die konkrete Festlegung erfolgt im Rahmen einer individuellen Risikoanalyse.

SYS.1.8.A21 Einsatz von Speicherpools zur Trennung von Mandanten (H)

Mandanten SOLLTEN Speicherressourcen aus unterschiedlichen sogenannten Speicherpools zugewiesen werden. Dabei SOLLTE ein Speichermedium immer nur einem einzigen Pool zugewiesen werden. Die logischen Festplatten (LUNs), die aus einem solchen Pool generiert werden, SOLLTEN nur einem einzigen Mandanten zugeordnet werden.

SYS.1.8.A22 Einsatz einer hochverfügbaren SAN-Lösung (H)

Eine hochverfügbare SAN-Lösung SOLLTE eingesetzt werden. Die eingesetzten Replikationsmechanismen SOLLTEN den Verfügbarkeitsanforderungen der Institution an die Speicherlösung entsprechen. Auch die Konfiguration der Speicherlösung SOLLTE den Verfügbarkeitsanforderungen gerecht werden. Außerdem SOLLTE ein Test- und Konsolidierungssystem vorhanden sein.

SYS.1.8.A23 Einsatz von Verschlüsselung für Speicherlösungen (H)

Alle in Speicherlösungen abgelegten Daten SOLLTEN verschlüsselt werden. Es SOLLTE festgelegt werden, auf welchen Ebenen (Data-in-Motion und Data-at-Rest) verschlüsselt wird. Dabei SOLLTE beachtet werden, dass die Verschlüsselung auf dem Transportweg auch bei Replikationen und Backup-Traffic relevant ist.

SYS.1.8.A24 Sicherstellung der Integrität der SAN-Fabric (H)

Um die Integrität der SAN-Fabric sicherzustellen, SOLLTEN Protokolle mit zusätzlichen Sicherheitsmerkmalen eingesetzt werden. Bei den folgenden Protokollen SOLLTEN deren Sicherheitseigenschaften berücksichtigt und entsprechende Konfigurationen verwendet werden:

- Diffie Hellman Challenge Handshake Authentication Protocol (DH-CHAP),
- Fibre Channel Authentication Protocol (FCAP) und
- Fibre Channel Password Authentication Protocol (FCPAP).

SYS.1.8.A25 Mehrfaches Überschreiben der Daten einer LUN (H)

In SAN-Umgebungen SOLLTEN Daten gelöscht werden, indem die zugehörigen Speichersegmente einer LUN mehrfach überschrieben werden.

SYS.1.8.A26 Absicherung eines SANs durch Hard-Zoning (H)

Um SANs zu segmentieren, SOLLTE Hard-Zoning eingesetzt werden.

4. Weiterführende Informationen

4.1. Wissenswertes

Die International Organization for Standardization (ISO) gibt in der Norm ISO/IEC 27040:2015 „Information technology – Security techniques – Storage security" Vorgaben für die Absicherung von Speicherlösungen.

Das Information Security Forum (ISF) macht in seinem Standard „The Standard of Good Practice for Information Security" im Kapitel SY1.4 Network Storage Systems Vorgaben für die Absicherung von Speicherlösungen.

Bundesamt
für Sicherheit in der
Informationstechnik

SYS.1.9 Terminalserver

1. Beschreibung

1.1. Einleitung

Ein Terminalserver ist ein Server, auf dem Client-Anwendungen (kurz Anwendungen) direkt ausgeführt werden und der nur deren grafische Oberfläche (Bedienoberfläche) an die Clients weiterleitet. Hierfür wird eine Terminalserver-Software verwendet. Der Terminalserver ist dann das zugrundeliegende IT-System, auf dem diese Software ausgeführt wird. Die Eingaben am Client, z. B. über Tastatur und Maus, werden an die Terminalserver-Software übertragen, die diese Eingaben dann dem Terminalserver übergibt. In der bereitgestellten Anwendung auf dem Terminalserver werden daraufhin die Aktionen ausgeführt, die gegebenenfalls durch die Eingaben ausgelöst werden und der Terminalserver ermittelt die neue (möglicherweise geänderte) Bedienoberfläche. Diese Bedienoberfläche wird dann von der Terminalserver-Software an den Client übertragen.

In einer Terminalserver-gestützten Umgebung verbinden sich typischerweise Clients mithilfe einer entsprechenden Terminal-Client-Software mit der Terminalserver-Software auf dem Terminalserver. Hierfür wird über Terminalserver-Protokolle kommuniziert, über die die Ein- und Ausgaben übertragen werden. Beispiele hierfür sind das Remote Desktop Protocol (RDP), Independent Computing Architecture (ICA), PC-over-IP (PCoIP) oder Virtual Network Computing (VNC).

Die Art der auf diese Weise bereitgestellten Anwendungen ist dabei prinzipiell nicht eingeschränkt und kann beispielsweise produktive Anwendungen wie Webbrowser, Office-Anwendungen oder Finanzsoftware, aber auch Administrationswerkzeuge wie SSH-Clients oder Management-Tools umfassen.

In einem typischen Einsatzszenario stellt ein Terminalserver mehreren Clients zentralisiert Anwendungen bereit, die aus organisatorischen oder technischen Gründen nicht lokal auf diesen Clients ausgeführt werden sollen oder können. Ein Beispiel hierfür sind Administrationstools, die nicht direkt auf den Clients der Administrierenden ausgeführt werden sollen. Ein weiteres Beispiel ist Software mit speziellen technischen Anforderungen an die zugrundeliegende Hardware der Clients, wie beispielsweise bestimmte Grafikkarten, die nicht auf allen Clients vorhanden sind.

In einer Terminalserver-gestützten Umgebung können die Clients sogenannte Fat Clients oder Thin Clients sein. Fat Clients sind mit einem vollwertigen Client-Betriebssystem ausgestattet. Thin Clients können dagegen nur genutzt werden, um sich mit dem Terminalserver zu verbinden und diesen zu bedienen.

Auf einem Terminalserver können mehrere Personen gleichzeitig auf demselben Betriebssystem arbeiten und dieselbe oder mehrere unterschiedliche Anwendungen parallel benutzen.

1.2. Zielsetzung

Ziel dieses Bausteins ist es, Informationen zu schützen, die beim Einsatz von Terminalservern gespeichert, verarbeitet und übertragen werden. Hierzu werden spezielle Anforderungen an die beteiligten Anwendungen, IT-Systeme und Netze gestellt.

1.3. Abgrenzung und Modellierung

Der Baustein SYS.1.9 *Terminalserver* ist sowohl auf den Terminalserver selbst als auch auf die zugreifenden Fat Clients und Thin Clients mit Terminal-Client-Software anzuwenden. Hierbei sind für Server und Clients jeweils sowohl die Soft- als auch die Hardwarekomponenten zu berücksichtigen.

Um ein IT-Grundschutz-Modell für einen konkreten Informationsverbund zu erstellen, muss grundsätzlich die Gesamtheit aller Bausteine betrachtet werden. In der Regel sind mehrere Bausteine auf das Thema bzw. Zielobjekt anzuwenden.

Dieser Baustein behandelt die folgenden Inhalte:

- Ein Terminalserver im Sinne dieses Bausteins ist jedes IT-System, auf dem Anwendungen auf die oben beschriebene Weise zentral zur Verfügung gestellt werden. Hierbei muss die Verbindung vom Client aus direkt initiiert werden.
- Der Baustein SYS.1.9 *Terminalserver* ist anzuwenden, wenn durch eine Terminal-Client-Software ausschließlich Eingaben der Benutzenden an den Terminalserver übermittelt werden.
- Dieser Baustein beinhaltet spezifische Anforderungen an die verwendeten Netze, um die Kommunikation zwischen Clients und Terminalserver abzusichern.

Folgende Inhalte sind ebenfalls von Bedeutung und werden an anderer Stelle behandelt:

- Für den Terminalserver und die Clients müssen die Bausteine SYS.1.1 *Allgemeiner Server* bzw. SYS.2.1 *Allgemeiner Client* sowie gegebenenfalls die spezifischen Bausteine für die Server- bzw. Client-Betriebssysteme angewendet werden.
- Auf die Terminalserver-Software müssen der Baustein APP.6 *Allgemeine Anwendung* sowie gegebenenfalls entsprechende weitere Bausteine der Schicht APP *Anwendungen* angewendet werden.
- Für die Anwendungen, die über den Terminalserver bereitgestellt werden, müssen zusätzlich der Baustein APP.6 *Allgemeine Anwendung* sowie gegebenenfalls die entsprechenden spezifischen Bausteine der Schicht APP *Anwendungen* angewendet werden.
- Der Baustein NET.1.1 *Netzarchitektur und –design* muss angewendet werden, um die für die Kommunikation zwischen Clients und Terminalserver verwendeten Netze abzusichern.

Dieser Baustein behandelt **nicht** die folgenden Inhalte:

- Fernwartungswerkzeuge sind keine Terminalserver im Sinne dieses Bausteins. Um diese Werkzeuge abzusichern, ist der Baustein OPS.1.2.5 *Fernwartung* umzusetzen.
- Wenn ein zu administrierendes IT-System über Terminalserver-Protokolle angesprochen wird, stellt dies keine Nutzung des Terminalservers im Sinne dieses Bausteins dar.
- Dieser Baustein adressiert nicht den Fall, dass Clients direkt auf andere Clients über Terminalserver-Protokolle oder Kollaborationswerkzeuge zugreifen.
- Falls der Terminalserver-Dienst über zusätzliche Sicherheitskomponenten wie Application Delivery Controller (ADC, siehe Kapitel 4 *Weiterführende Informationen*) zur Verfügung gestellt wird, sind diese zusätzlichen Komponenten gesondert zu betrachten.

2. Gefährdungslage

Da IT-Grundschutz-Bausteine nicht auf individuelle Informationsverbünde eingehen können, werden zur Darstellung der Gefährdungslage typische Szenarien zugrunde gelegt. Die folgenden spezifischen Bedrohungen und Schwachstellen sind für den Baustein SYS.1.9 *Terminalserver* von besonderer Bedeutung.

2.1. Qualitätsverlust der Anwendungsbereitstellung

Eine vom Terminalserver bereitgestellte Anwendung wird in Echtzeit genutzt. Da die Bedienoberfläche auf dem Terminalserver vorbereitet und an die Clients übertragen wird, kann nur reibungslos gearbeitet werden, wenn die Antwort des Terminalservers auf eine Eingabe ohne merkliche Zeitverzögerung und klar erkennbar bei den Clients ankommt. Empfangen die Clients die Antworten des Terminalservers zeitlich verzögert, kann die Bedienbarkeit soweit einschränkt sein, dass dies einem Ausfall des Dienstes gleichkommt. Sowohl eine konstant hohe Verzögerung als auch häufig auftretende und nicht vorhersehbare Spitzen können diesen Effekt hervorrufen.

Eine zu hohe Verzögerung kann durch eine zu hohe Latenz der Übertragungsstrecken oder Netzkomponenten hervorgerufen werden. Wird die Kommunikation beispielsweise über weitere Sicherheitskomponenten wie VPN-Gateways abgesichert, die möglicherweise unzureichend dimensioniert sind, kann die Verzögerung weiter erhöht werden. Dies kann dazu führen, dass die Anwendung nur noch eingeschränkt genutzt werden kann.

Ist der Terminalserver stark ausgelastet, kann dieser nur verzögert antworten. Ist beispielsweise die CPU oder der Arbeitsspeicher unzureichend dimensioniert, kann der Terminalserver schnell überlasten und schließlich nur verzögert antworten. Ähnliches gilt, wenn der Terminalserver von zu vielen Personen zeitgleich genutzt wird.

Ist der Bildschirminhalt nicht klar genug erkennbar, kann mit dem Terminalserver nicht mehr effizient gearbeitet werden. Beispielsweise können Schrift oder Mauszeiger aufgrund von Kompressionsartefakten schwer zu erkennen sein, falls nicht genügend Leitungskapazität zur Verfügung steht.

All dies kann dazu führen, dass die Benutzenden entweder nicht oder nur noch stark eingeschränkt den Terminalserver nutzen können.

2.2. Ausfall der Anwendungsbereitstellung

In einer Terminalserver-gestützten Umgebung werden Anwendungen zentral ausgeführt und deren Ausgabe an die entsprechenden Clients übertragen. Ist der Terminalserver nicht verfügbar, können keine Eingaben mehr verarbeitet werden und die von dem Terminalserver bereitgestellten Anwendungen versagen unmittelbar ihren Dienst. Beziehen die Clients ihre gesamte Bedienoberfläche vom Terminalserver, fällt aus der Perspektive der Benutzenden das IT-System vollständig aus.

Wenn der Client ausfällt, kann hierüber nicht auf die vom Terminalserver bereitgestellten Anwendungen zugriffen werden, auch wenn diese dort verfügbar sind. Ähnliches gilt, wenn die Verbindung zwischen Client und Terminalserver gestört ist.

Von Ausfällen des Netzes oder des Terminalservers sind in der Regel nicht nur einzelne Clients betroffen. In vielen Fällen sind zahlreiche oder sogar alle Clients einer Institution auf den Terminalserver angewiesen. Fällt der Terminalserver aus, ist in diesem Fall eine große Anzahl von Clients gleichzeitig betroffen.

2.3. Unzureichende Netztrennung für Terminalserver

Auf Terminalservern werden meist Anwendungen bereitgestellt, die als Client fungieren. Hierdurch ähnelt ein Terminalserver von der Vertrauenswürdigkeit her eher einem Client als einem Server.

Wird dies bei der Netztrennung nicht geeignet berücksichtigt, kann unter Umständen über den Terminalserver unberechtigterweise auf weitere Serveranwendungen zugegriffen werden, beispielsweise über einen Webbrowser. Hierdurch kann der Terminalserver als Ausgangspunkt für Angriffe auf weitere IT-Systeme und Anwendungen missbraucht werden.

Durch die Eingaben am Client ist ein hoher Grad an Interaktion mit dem Terminalserver zu erwarten. Hierdurch können mögliche Schwachstellen leichter ausgenutzt werden. Dies ist insbesondere dann relevant, wenn ein Terminalserver Anwendungen für Benutzendengruppen bereitstellt, die unterschiedlichen Netzsegmenten zugeordnet sind. In diesem Fall könnte vom Terminalserver aus unautorisiert auf weitere Anwendungen in diesen Netzsegmenten zugegriffen werden.

2.4. Unzureichende Absicherung von Sitzungen auf dem Terminalserver

Terminalserver können unterschiedlichen Clients dedizierte Anwendungsinstanzen bereitstellen, die auf demselben Betriebssystem ausgeführt werden. Dabei teilen sich die Anwendungen unter anderem gemeinsam genutzte Bibliotheken, den Kernel und die benötigten Ressourcen des Terminalservers (z. B. CPU oder RAM).

Aufgrund von Fehlkonfigurationen oder Software-Schwachstellen können einzelne Anwendungsinstanzen gegebenenfalls miteinander kommunizieren, ohne dass dies ursprünglich vorgesehen war. Werden beispielsweise Sitzungen auf Terminalservern mit zu weitreichenden Berechtigungen ausgeführt, kann unter Umständen aus einer Anwendung heraus auf beliebige Teile des Dateisystems zugegriffen werden. Dies kann beispielsweise über Programmdialoge zum Speichern oder Öffnen von Dateien ausgenutzt werden, über die nicht vorgesehene Bereiche der Festplatte beschrieben oder gelesen werden.

Ein weiteres Beispiel ist das sogenannte RDP Session Hijacking, das auf den Sitzungen des Terminalservers selbst beruht. Bleiben Benutzende weiterhin angemeldet, nachdem ihre Sitzungen am Terminalserver beendet sind, kann dies zu Problemen führen. Sind Angreifende mit entsprechenden Rechten ausgestattet, die sie beispielsweise durch ein unzureichendes Rechtemanagement oder durch Ausnutzen von Software-Schwachstellen zuvor erhalten haben, können sie unter Umständen aus einer anderen Sitzung heraus eine bestehende Sitzung übernehmen. In diesem Fall können Angreifende die Sitzung im Kontext des oder der Benutzenden fortsetzen.

Wird das Betriebssystem von mehreren Anwendungen oder Anwendungsinstanzen gemeinsam genutzt, können gegebenenfalls Sitzungen anderer Benutzenden über CPU oder RAM beeinflusst werden. Hierfür müssen in den entsprechenden Anwendungen entsprechende Sicherheitslücken vorhanden sein, über die die benötigte Schadsoftware ausgeführt werden kann. Beispielsweise kann dann eine spezielle Schadsoftware Passwörter aus dem RAM auslesen. Auch ohne Sicherheitslücken in der Software können durch Sicherheitslücken in der Hardware (z. B. Meltdown) Angreifende beliebige schützenswerte Daten anderer Sitzungen auslesen.

2.5. Unzureichende Absicherung des Terminalserver-Protokolls

Viele Terminalserver-Protokolle bieten die Möglichkeit einer authentisierten und verschlüsselten Kommunikation. Diese Möglichkeit ist jedoch nicht immer ausreichend, um die Kommunikation abzusichern. Nutzt das Terminalserver-Protokoll veraltete und angreifbare Mechanismen oder werden durch Fehlkonfigurationen wichtige Sicherheitsfunktionen abgeschaltet, kann die Kommunikation zwischen Clients und Terminalserver abgehört werden. Informationen, die zwischen dem Terminalserver und den Clients übertragen werden und unter Umständen abgehört oder verändert werden, sind insbesondere:

- Authentisierungsinformationen und Eingaben von Benutzenden, die von den Clients zu den Terminalservern gesendet werden,
- Bildschirminformationen, die auf den Clients ausgegeben werden,
- Daten aus der Zwischenablage,
- Dateitransfers zwischen den lokalen Laufwerken des Clients und dem Server sowie
- Informationen von umgeleiteten Geräten des Clients (z. B. Audiogeräte, serielle- oder parallele Schnittstellen, USB-Geräte und Drucker).

Aber auch wenn die Protokollmechanismen die Kommunikation grundsätzlich stark genug absichern, kann die Implementierung des Protokolls innerhalb eines Terminalservers oder einer Terminal-Client-Software Schwachstellen beinhalten. Dies kann dazu führen, dass der Terminalserver direkt angreifbar wird, ohne dass die Kommunikation ausgespäht werden muss.

2.6. Unberechtigte Nutzung von Sammelkonten

Falls mehrere Personen eine Anwendung auf einem Terminalserver zu unterschiedlichen Zeiten nutzen wollen, werden oft Sammelkonten eingerichtet. Dies kann jedoch internen Regelungen oder den Lizenzbedingungen der über den Terminalserver bereitgestellten Software widersprechen.

Werden Sammelkonten verwendet, verhindert dies auch, dass die auf dem Terminalserver ausgeführten Aktionen konkreten Personen zugeordnet werden können. Hierdurch kann nicht mehr nachvollzogen werden, wer was getan hat. Dies kann insbesondere ein rechtliches Risiko bedeuten, wenn es gesetzliche Anforderungen an die Nachvollziehbarkeit gibt, z. B. falls auf dem Terminalserver personenbezogene Daten verarbeitet werden.

2.7. Ungeeignete Einschränkung der Zugriffsrechte der Benutzenden

Ein Terminalserver kann zeitgleich sowohl als Server und bezogen auf die auf ihm ausgeführten Anwendungen auch als Client fungieren. Dies kann zu Fehlern in der Vergabe der Zugriffsrechte führen.

Sichere Konfigurationen von IT-Systemen und Anwendungen sehen zumeist eine möglichst restriktive Rechtevergabe vor. Dies gilt insbesondere auch für Terminalserver. Werden die Berechtigungen für die Benutzung eines Terminalservers jedoch zu weit eingeschränkt, können die Benutzenden die bereitgestellten Anwendungen nur noch stark eingeschränkt nutzen. Dies kann sowohl aus einer zu strengen Richtlinie als auch aus einer Fehlkonfiguration resultieren.

Wird die Arbeit durch solche Einschränkungen zu sehr erschwert, indem beispielsweise der Schreibzugriff auf lokale Laufwerke komplett verboten wird, kann dies unerwünschte Auswirkungen haben. Beispielsweise könnten Benutzende auf nicht vorgesehene Workarounds ausweichen und Daten z. B. nach einem Export über Datenaustauschplattformen an ungeeigneter Stelle verarbeiten.

2.8. Ungeeignete Anwendungen für den Einsatz auf Terminalservern

Nicht alle Anwendungen lassen sich auf beliebigen Terminalservern bereitstellen. Werden beispielsweise notwendige Funktionen der Graphics Processing Unit (GPU) in der emulierten Grafikeinheit nicht unterstützt, können 3D-Anwendungen über einen Terminalserver nicht oder nur eingeschränkt genutzt werden. Ähnliches gilt, wenn Eingaben von anwendungs- oder branchenspezifischen Peripheriegeräten vom Terminalserver, der Terminal-Client-Software oder dem Terminalserver-Protokoll nicht unterstützt werden.

Werden einzelne Anwendungsfunktionen oder die Anbindung von Peripheriegeräten vor der Beschaffung nicht oder nur unzureichend getestet, werden diese Einschränkungen möglicherweise erst im laufenden Betrieb festgestellt. Dadurch kann die Verfügbarkeit der Anwendung erheblich eingeschränkt sein und der Terminalserver kann möglicherweise nicht wie vorgesehen eingesetzt werden. Gegebenenfalls muss er sogar komplett ersetzt werden.

3. Anforderungen

Im Folgenden sind die spezifischen Anforderungen des Bausteins SYS.1.9 *Terminalserver* aufgeführt. Der oder die Informationssicherheitsbeauftragte (ISB) ist dafür zuständig, dass alle Anforderungen gemäß dem festgelegten Sicherheitskonzept erfüllt und überprüft werden. Bei strategischen Entscheidungen ist der oder die ISB stets einzubeziehen.

Im IT-Grundschutz-Kompendium sind darüber hinaus weitere Rollen definiert. Sie sollten besetzt werden, insofern dies sinnvoll und angemessen ist.

Zuständigkeiten	Rollen
Grundsätzlich zuständig	IT-Betrieb
Weitere Zuständigkeiten	Planende

Genau eine Rolle sollte *Grundsätzlich zuständig* sein. Darüber hinaus kann es noch *Weitere Zuständigkeiten* geben. Falls eine dieser weiteren Rollen für die Erfüllung einer Anforderung vorrangig zuständig ist, dann wird diese Rolle hinter der Überschrift der Anforderung in eckigen Klammern aufgeführt. Die Verwendung des Singulars oder Plurals sagt nichts darüber aus, wie viele Personen diese Rollen ausfüllen sollen.

3.1. Basis-Anforderungen

Die folgenden Anforderungen MÜSSEN für diesen Baustein vorrangig erfüllt werden.

SYS.1.9.A1 Erstellung einer Sicherheitsrichtlinie für den Einsatz von Terminalservern (B)

Für den Einsatz von Terminalservern MUSS eine Sicherheitsrichtlinie erstellt werden. Bei der Erstellung der Sicherheitsrichtlinie MÜSSEN mindestens folgende Punkte berücksichtigt werden:

- Anwendungen, die auf Terminalservern bereitgestellt werden dürfen,
- Anwendungen, die gemeinsam auf Terminalservern bereitgestellt werden dürfen,
- Anforderungen an die Sicherheit von Clients, auf denen die Terminal-Client-Software ausgeführt wird,
- physisches Umfeld, in dem die Clients eingesetzt werden dürfen,
- Netze, aus denen heraus Kommunikationsverbindungen zu den Terminalservern initiiert werden dürfen,
- Netze, in die Anwendungen auf den Terminalservern kommunizieren dürfen,
- Kommunikationsprotokolle, die zwischen Clients und Terminalservern erlaubt sind,
- Verschlüsselungsmechanismen und Authentisierungsmethoden, die zwischen Clients und Terminalservern zu benutzen sind,

- Möglichkeiten, wie Dateien und Anwendungsdaten zusätzlich zur Bildschirmausgabe über das Terminalserver-Protokoll übertragen werden dürfen sowie
- Peripheriegeräte, die neben Ein- und Ausgabegeräten zusätzlich an den Client angebunden werden dürfen.

SYS.1.9.A2 Planung des Einsatzes von Terminalservern (B)

Für die Anwendungen, die auf einem Terminalserver bereitgestellt werden sollen, MÜSSEN die Anforderungen an die Funktionalität (Anforderungsprofil) ermittelt werden. Für alle benötigten Funktionen MUSS sichergestellt werden, dass diese tatsächlich auch über den Terminalserver abgerufen werden können. Darüber hinaus MUSS getestet werden, ob die Anwendungen die Anforderungen bei der Bereitstellung über den Terminalserver grundlegend erfüllen.

Die Gesamtzahl der einzurichtenden Benutzenden MUSS prognostiziert werden. Dabei MÜSSEN alle Anwendungen mitgezählt werden, die auf dem Terminalserver bereitgestellt werden.

Die Anzahl der Benutzenden, die den Terminalserver potenziell gleichzeitig benutzen, MUSS prognostiziert werden. Diese Prognosen MÜSSEN den Einsatzzeitraum des Terminalservers abdecken.

Abhängig von der prognostizierten Anzahl der Benutzenden und den Anforderungen der bereitgestellten Anwendungen MÜSSEN die Leistungsanforderungen (z. B. hinsichtlich CPU und Arbeitsspeicher) an den Terminalserver ermittelt werden. Der Terminalserver MUSS anhand dieser Leistungsanforderungen dimensioniert und ausgestattet werden.

Das Lizenzschema der eingesetzten Anwendungen MUSS daraufhin geprüft werden, ob es dafür geeignet ist, diese Anwendungen auf Terminalservern einzusetzen.

SYS.1.9.A3 Festlegung der Rollen und Berechtigungen für den Terminalserver (B)

Auf Terminalservern DÜRFEN KEINE Sammelkonten verwendet werden, wenn dies gegen interne Regelungen oder Lizenzbedingungen verstößt. Bei der Festlegung von Rollen und Berechtigungen für die Benutzung des Terminalservers MÜSSEN alle auf dem Terminalserver bereitgestellten Anwendungen und deren Anforderungen mit ausreichenden Berechtigungen ausgestattet werden.

Die Rollen und Berechtigungen MÜSSEN so vergeben werden, dass zwischen Terminalserver-Sitzungen nur in dem Umfang kommuniziert werden kann, wie es für die Funktionalität der Anwendung erforderlich ist. Mindestens MÜSSEN die Berechtigungen für folgende Tätigkeiten festgelegt werden:

- Ausführen von Anwendungen in fremdem Kontext (insbesondere als „root“),
- Zugriff auf betriebssystemspezifische Funktionen,
- Zugriff auf das Dateisystem des Terminalservers,
- Zugriff auf Schnittstellen und Dateisystem des verwendeten zugreifenden Clients,
- Zugriff der auf dem Terminalserver bereitgestellten Anwendungen auf nachgelagerte Dienste,
- Datei- und Objekttransfer zwischen Clients und Terminalservern (z. B. zum Drucken am Client) sowie
- Anbindung von Peripheriegeräten am Client.

SYS.1.9.A4 Sichere Konfiguration des Terminalservers (B)

Abhängig von den Anforderungen an die Sicherheit und Funktionalität der bereitgestellten Anwendungen MÜSSEN Vorgaben für die Konfiguration von Terminalservern erstellt werden. Diese Vorgaben MÜSSEN vollständig umgesetzt und dokumentiert werden.

Es MUSS geprüft werden, ob das Unternehmen, das den Terminalserver herstellt, Vorgaben oder Empfehlungen zur sicheren Konfiguration oder Härtung bereitstellt. Ist dies der Fall, MÜSSEN diese für die Erstellung der Konfigurationsvorgaben angemessen berücksichtigt werden. Sowohl die Konfigurationsvorgaben als auch deren Umsetzung MÜSSEN regelmäßig geprüft und gegebenenfalls angepasst werden.

Es MÜSSEN mindestens folgende Punkte für die Konfigurationsvorgaben berücksichtigt werden:

- Rollen und Berechtigungen
- Umfang der Verschlüsselung des Terminalserver-Protokolls